Windesheim

Controlling the COA framework

Thesis

Contact information

Author:

Name:	: Jeroen J. Willemsen
Education	: Business Informatics
E-mail	: jeroenwillemsen2001@gmail.com
Phone number	: 06-2412 64 85
Address	: Linneausparkweg 184 1098 EP Amsterdam
Duration of internship	: 14-03-08 to 14-11-08

External supervisor

Name	: Drs. Marco Plas
E-mail	: marco.plas@domustechnica.nl
Telephone	: 06-1503 04 13

Organisational Supervisor

Name	: Drs. Alina Stan
E-mail	: alina.stan@capgemini.com
Telephone	: 06-4589 82 04

College Supervisor

Name	: P. Phillipsen
E-mail	: pra.philipsen@windesheim.nl
Phone number	: 038-469 99 11

Organisation

Name	: Capgemini Nederland bv.
Division	: F55/AGI : Architecture, Governance & Infrastructure
Telephone	: 030-698 00 00
Address	: Papendorpseweg 100
PO Box	: Postbus 2575, 3500 GN, Utrecht
Website	: http://www.capgemini.com

College Institute

Institute name	: Christelijke Hogeschool Windesheim
Academy	: School of Information Science
Telephone	: 038-469 99 11
Address	: Campus 6-10
Website	: http://www.windesheim.nl

Windesheim

Abstract – Controlling the COA framework

The evolving enterprise of today

Enterprise 2.0, deconstructed firms, organisational ecosystems: all of these buzzwords are used to identify operating concepts behind today's companies. They are driven by the need to be agile, flexible, distributed, open, transparent, simple, on demand and with a short time to market.

In order to be all this, we have decided to break down our organisations, outsource non-core activities and create relationships with vendors, clients, resellers, producers and other parties. Many of these relationships extend company boundaries. We have globalised our network in order to survive and thrive on the international markets. We try to collaborate with those that will provide us the greatest benefits for the smallest risk and price. We are not the only ones. Our competitors do it as well. We all have created vast networks in which we can often find our own competitors as a part of it. In fact, some of us will even directly work together with them.

Relationships provide both the key as well as certain issues

These relationships are based on collaboration, which automatically implicates sharing information and sometimes even services, products and processes. Such activities allow the business ecosystems to obtain many benefits in terms of scale, lower costs, better innovative capacities, knowledge channels, reduction of uncertainty, et cetera.

Yet, those benefit-providing relationships create many issues that will trouble the collaborating firms and their management on both ends of the deal. Whenever we want to collaborate across a national border, we have to take legislation of at least two countries in mind. For instance, if you are a European company, you have to take "Code Tabaksblat" in mind. If American, there is "SOX" to consider. Now if you want to collaborate with both an American and a European enterprise, then both laws must be followed. So, what about security and compliance? How can we be sure that our collaborative partners will be trustworthy and treat our information with the same policies as we do? How can we be sure that we follow the right directions in such relations? No one wants to be sued because of violating such agreements.

Then there is the demand for connectivity: we need to communicate with our partners. How can we do this safely and securely? How can we make sure that these parties will be able to see important information, while others should not have that ability?

Today's solution:

To face these issues, many came up with different solutions. SOA-based frameworks for collaboration, SLA's, Enterprise 2.0, automated distributed compliancy systems, ideas for a de-perimeterised environment. Most of these answers proved to be quite a step forward to a secured dynamic collaborative relationship. Yet all of them lack a fully integrated set of mechanisms to tackle the sum of all problems.

Tomorrow's solution:

Luckily, progression does not stop here. One of the biggest breakthroughs is nearby: the creation of the Collaboration Oriented Architectures (COA). At this moment, many details around the concept of the COA are unknown or not yet fully developed.

In order to accelerate these developments and reveal many details around the concept, a research project is created around the question:

"What is the Collaboration Oriented Architecture and how can it be used?"

The project is executed by the Security and Innovation Research Centre in Capgemini[1] NL, in collaboration with Windesheim[2], Eli Lilly[3] and the Jericho Forum[4], in order to create the first information architecture that is based on SOA and the concept of de-perimeterisation, which will allow us to work together dynamically and safely with whoever we choose.

[1] Capgemini : www.capgemini.com , visited at 20-10-08.

[2] Windesheim: www.windesheim.nl , visited at 20-10-08.

[3] Eli Lilly: http://www.lilly.com/ , visited at 20-10-08.

[4] The Jericho Forum: www.theopengroup.org/jericho , visited at : 20-10-08.

Guide to Reader

If you are interested in:

- Service Oriented Architecture as a teacher or a student: read paragraph 2.2.
- Collaboration theories, read paragraph 2.5 and chapter 3.
- IT-governance, read paragraph 2.4.
- The Collaboration Oriented Architectures, read chapter 3.
- The Jericho Forum and its concepts, read paragraph 2.6 and chapter 3.
- This thesis, but know not what to expect, read the abstract and chapters 1 and 4.
- Interested in the new detailing of the COA framework as a Jericho Specialist: read chapters 1, 3 and 4.
- Interested in this thesis and have only limited time, read the abstract and chapters 1 and 4.
- Interested in this thesis and have nothing but time as an evaluating teacher, friend, family member, read all chapters and the abstract.

If you do not have much time on your hands, you can stick to reading the introduction and the conclusions. Each chapter and paragraph has an introducing and concluding/summarising section. Each larger section has its own introduction at the beginning and conclusion and/or summary at the end of the section.

COA v2.0 compatibilty

This research has been conducted while COA v2.0 was in development. However it is still usable and compatible with the v2.0 release. Additional references have been included in section 2.6 and chapter 3 to related materials which have been released after this research.

Abstract Table of Contents

Abstract – Controlling the COA framework 3

Guide to Reader 5

Abstract Table of Contents 6

Full Table of Contents 7

Table of Figures 10

Table Overview 11

1. Introduction to the thesis and Subjects 12

2. Background 17

3. The COA framework 1

4. Conclusions and Recommendations 269

Literature 283

Appendices 289

Appendix A1: The 11 Jericho Commandments 290

Appendix A2: Jericho Commandments explained 291

Appendix A3: Jericho Roadmap 297

Windesheim

Full Table of Contents

Abstract – Controlling the COA framework 3

Guide to Reader 5

COA v2.0 compatibilty 5

Abstract Table of Contents 6

Full Table of Contents 7

Table of Figures 10

Table Overview 11

1. Introduction to the thesis and Subjects 12

1.1. Introduction 12
1.2. Overview 14
1.3. Research goal 15
1.4. Problem definition and research questions 15
1.4.1. Problem definition 15
1.4.2. Research questions 15
1.5. Thesis Outline 16
1.6. Research Plan 16
1.7. Relevance 16

2. Background 17

2.1. Introduction 17
2.2. Introduction to Service Oriented Architecture 18
2.2.1. Introduction 18
2.2.2. Service Oriented Architecture 18
2.2.3. SOA and Security 19
2.2.4. SOA concepts 33
2.2.5. SOA and the Collaboration Oriented Architecture: Why is it important? 35
2.2.6. Concluding: What is SOA and where do we focus on? 35
2.3. Introduction to Software as a Service 37
2.3.1. Introduction 37
2.3.2. Definitions: SaaS as we see it 37
2.3.3. SaaS and security 38
2.3.4. Concluding: What is SaaS and where do we focus on? 40
2.4. Control Objectives for Information and related Technology 41
2.4.1. Introduction 41
2.4.2. IT Governance: 41
2.4.3. Advantages of IT Governance: 45
2.4.4. IT Governance focus: 45
2.4.5. COBIT: 46
2.4.6. Use of COBIT for this thesis 50
2.4.7. Concluding: What is COBIT and where do we focus on? 51
2.5. Collaboration 54
2.5.1. Introduction 54
2.5.2. What is collaboration? 54
2.5.3. Benefits and importance of Collaboration 56
2.5.4. Roles and relations in collaboration 57

2.5.5. Collaboration and Prisoners' Dilemma *60*
2.5.6. Internet business models *61*
2.5.7. The collaborative landscape *63*
2.5.8. Concluding: Collaboration and the focus for this thesis *63*
2.6. The Jericho Forum and its concepts 65
2.6.1. Introduction and overview *65*
2.6.2. The Jericho Forum *65*
2.6.3. De-perimeterisation explained *67*
2.6.4. Surviving in a hostile world by using open and inherently secure standards, systems, and protocols *70*
2.6.5. Policy management *81*
2.6.6. Data classification, protection and privacy issues *94*
2.6.7. Identity Management, user authentication and federation *105*
2.6.8. Trust, Trustmanagement and Trust brokers *119*
2.6.9. End-point security *134*
2.6.10. IT-Audit *141*
2.6.11. Summarising: the Jericho concept *146*
2.7. Summary and overview 149

3. The COA framework 1

3.1. Introduction 150
3.2. Looking back: the current situation with its complications 152
3.2.1. Introduction *152*
3.2.2. The current situation in the markets of today *152*
3.2.3. The current status of the Jericho concepts *153*
3.2.4. Concluding: the necessity of the framework *155*
3.3. An introduction to the COA framework: general overview 155
3.3.1. Introduction *155*
3.3.2. The Architects' View and its components *156*
3.3.3. Framework Purpose in short *1*
3.3.4. How to adopt the framework in short *163*
3.3.5. Summary: The Collaboration Oriented Architecture Framework in short *163*
3.4. Main Value: SLATES, collaborative possibilities and security 164
3.4.1. Introduction *164*
3.4.2. SLATES *164*
3.4.3. Collaborative possibilities *168*
3.4.4. Value to de-perimeterisation and information security *169*
3.4.5. Summary: the importance of the COA framework *169*
3.5. COA principles 170
3.5.1. Introduction *170*
3.5.2. The principles explained *170*
3.5.3. Summary: the COA Principles *176*
3.6. COA processes 180
3.6.1. Introduction *180*
3.6.2. People Lifecycle Management *181*
3.6.3. Risk Management *186*
3.6.4. Information Lifecycle Management *189*
3.6.5. Device Lifecycle Management *196*
3.6.6. Enterprise Relationship Management *203*
3.6.7. Summary: the COA Processes *211*
3.7. COA services 212
3.7.1. Introduction *212*
3.7.2. Identity Management, Federation and Reputation *212*
3.7.3. Trust Management and Classification *215*

3.7.4. Policy Management *224*
3.7.5. Meta/Information Management *227*
3.7.6. Audit *231*
3.7.7. Summary: the services defined in the COA framework *234*
3.8. COA quality attributes 235
3.8.1. Introduction *235*
3.8.2. The Quality attributes: description and measurements *235*
3.8.3. Summary: the COA Quality Attributes *239*
3.9. COA technologies 239
3.9.1. Introduction *240*
3.9.2. COA Technologies *240*
3.9.3. Summary: COA Technologies *241*
3.10. Intermezzo: Mapping the COA framework to paragraph 2.6 and the commandments 243
3.11. Application and adoption of the COA framework 245
3.11.1. Introduction *245*
3.11.2. Considering the COA framework and SaaS *246*
3.11.3. Implementing COA framework elements in an enterprise architecture *252*
3.11.4. Implementing COA framework elements in a network architecture *257*
3.11.5. Implementing COA framework elements in a system architecture *258*
3.11.6. Summarising: the adoption of the COA framework *261*
3.12. Summary: The COA framework 262

4. Conclusions and Recommendations 269

4.1. Introduction 269
4.2. Conclusions 269
4.2.1. Introduction *269*
4.2.2. The COA framework defined *269*
4.2.3. The importance of the COA framework *273*
4.2.4. The application or adoption of the COA framework *273*
4.2.5. The COA framework and its usage *276*
4.3. Recommendations 277
4.3.1. Introduction *277*
4.3.2. Recommendations related to the COA framework Processes *277*
4.3.3. Recommendations related to the COA framework Services *278*
4.3.4. Recommendations related to the COA framework application *280*
4.4. Further research 281

Literature 283

Appendices 289

Appendix A1: The 11 Jericho Commandments 290

Appendix A2: Jericho Commandments explained 291

Appendix A3: Jericho Roadmap 297

Table of Figures

Figure 1: Relationships between the different elements....14
Figure 2: Research path....16
Figure 3: Relationship between the different elements, marked areas will be covered in this chapter....17
Figure 4: The Web Services Security Stack in context (Jothy Rosenberg 2004)....22
Figure 5: IT Governance focuses. (Institute 2007a)....45
Figure 6: The main principle of COBIT (Institute 2007a)....46
Figure 7: The COBIT Framework. (Institute 2007a)....48
Figure 8: Mapping between PDCA and COBIT domains....49
Figure 9: Overall COBIT Framework. (Institute 2007a)....53
Figure 10: A model of Collaboration types, based on (Henry Mintzberg 1996)....54
Figure 11: Deconstructed firm versus integrated firm. (Joziasse 2008)....54
Figure 12: Different relations. (Joziasse 2008)....56
Figure 13: Organisational responses to stakeholder pressures. (Joziasse 2008)....56
Figure 14: Strategic meaning of the relationship. (Joziasse 2008)....57
Figure 15: The collaborative landscape. (Ralph Welborn 2008)....63
Figure 16: The different levels of protection mechanisms. (Stan 2008a)....72
Figure 17: Typical Proxy-agents. (John Wack 2002)....76
Figure 18: Automated Security Classification model. (Clark 2008)....95
Figure 19: Mapping trust levels and traffic light protocol. (Seccombe 2007))....96
Figure 20: Data centric security. (Forum 2007b)....99
Figure 21: Identity 1.0. (Hardt 2005)....104
Figure 22: Identity 2.0. (Hardt 2005)....104
Figure 23: Identity 2.0 with multiple credential providers. (Hardt 2005)....104
Figure 24: Digital Identity Lifecycle (based on (Barannikov 2008))....109
Figure 25: Overview of trust models. (Sabater 2005)....123
Figure 26: Trust architecture. (Forum 2006f)....124
Figure 27: The trust broker as proposed in. (Forum 2006f)....124
Figure 28: Current situation. (Arnold 2008)....132
Figure 29: Peer to Peer....134
Figure 30: Hybrid Peer to Peer....135
Figure 31: Trust broker....135
Figure 32: Relationship between the different elements, marked areas have been covered in this section....146
Figure 33: Thesis structure, marked area will be covered in this section. Marked transparent areas have been covered....147
Figure 34: Overview of framework detailing process....151
Figure 35: Overview of paragraph 3.3....152
Figure 36: The Collaboration Oriented Architecture Framework from an architects' view (based on (Forum 2008e))....156
Figure 37: Principles of the COA framework....154
Figure 38: Processes of the COA framework....155
Figure 39: People Lifecycle Management (simplified)....156
Figure 40: Risk Management (simplified)....156
Figure 41: Information Lifecycle Management (simplified)....156
Figure 42: Device Lifecycle Management (simplified)....157
Figure 43: Enterprise Relationship Management (simplified)....157
Figure 44: Services of the COA framework....157
Figure 45: Attributes of the Solution (part of the COA framework)....158
Figure 46: Technologies (fifth group of the COA framework)....159
Figure 47: Overview of paragraph 3.5, 3.6, 3.7, 3.8 and 3.9....170
Figure 48: Principles of the COA framework....167

Figure 49: COA framework processes. 176
Figure 50: Relations between People Lifecycle Management, SLATES, Principles and Services. 181
Figure 51: People Lifecycle Management. 178
Figure 52: Relations between Risk Management, SLATES, Principles and Services. 186
Figure 53: Risk Management (simplified) 183
Figure 54: Relations between Information Lifecycle Management, SLATES, Principles and Services. 189
Figure 55: Information Lifecycle Management (simplified). 186
Figure 56: Relations between Device Lifecyle Management, SLATES, Principles and Services. 196
Figure 57: Device lifecycle management (simplified). 193
Figure 58: Relations between Enterprise Relationship Management, SLATES, Principles and Services. 204
Figure 59: Enterprise relationship management (simplified). 201
Figure 60: The COA framework Services. 208
Figure 61: Digital Identity Lifecycle (based on (Barannikov 2008)) 208
Figure 62: Attributes of the Solution (part of the COA framework). 229
Figure 63: Technologies. 234
Figure 64: Overview of paragraph 3.11, detailing the COA framework application. 245
Figure 65: Distribution model 'Integration as a Service': distributive parties. (Dirk Hanenberg 2008) 241
Figure 66: Distribution model 'Integration as a Service': flow of software services / data. (Dirk Hanenberg 2008) 242
Figure 67: the ADM cycle. (OpenGroup 2005) 248
Figure 68: Thesis structure. COA related areas have been covered in this chapter. 257
Figure 69: Processed steps of the thesis. 263
Figure 70: Thesis structure, transparent areas have been covered. 264
Figure 71: The Collaboration Oriented Architecture Framework from an architects' view (based on (Forum 2008e)) 265

Table Overview

Table 1: Mapping between JFC's and COBITs Control Objectives 51
Table 2: Mapping Security Services and – Mechanisms. (Stan 2008a) 71
Table 3: Overview of message protection. (Stan 2008a) 78
Table 4: Mapping of trust broker functionalities and current existing technologies, part 1. . 131
Table 5: Comparison application SLATES in the consumer and the current enterprise market. (source: interview with A. Seccombe (April 2008), CSO Elli Lilly) 163
Table 6: relationships between the COA Principles, part one. 178
Table 7: Relationships between the COA Principles, part two. 179
Table 8: Relationships between the COA Principles, part three. 180
Table 9: Summary of People Lifecycle Management processes. 184
Table 10: Indication of Risk Management processes 188
Table 11: Processes description of Information Lifecycle Management. 193
Table 12: Processes description of Device Lifecycle Management. 200
Table 13: Processes description of Enterprise Relationship Management. 208
Table 14: Mapping between COA framework elements, paragraph 2.6 and the Jericho forum commandments, part 1. 243
Table 15: Mapping between COA framework elements, paragraph 2.6 and the Jericho forum commandments, part 2. 244

1. Introduction to the thesis and Subjects

1.1.Introduction

Due to disruptive technologies, such as the Internet, many markets went through important changes (Ralph Welborn 2008). New ways to facilitate the business have been introduced, such as high-speed communication channels, portals, online auctions, communities, enterprise resource planning software.
Companies started using the Internet as a medium for looking at their value chain. instead of their own processes. This was done to further optimise the complete chain, from acquiring basic resources to delivering the final product to an ever more demanding client. By doing so, it became possible to achieve multiple goals like generating revenue, creating or finding new business opportunities and introducing new business models that focus on value exchange. (Boonstra 2002) Because most companies did this, the market grew more competitive than before. In order to nowadays survive and thrive, being capable to compete and to gain a better market share, a new answer is necessary. One needs to innovate and find new needs of the client. (Philip Kotler 2002) However, with this many parties involved trying to innovate and change their position in the market, that market has become an unpredictable place for business. (Ralph Welborn 2008)
In order to face the new threats[5] and uncertainties and to get the best of every opportunity, a fast dynamic way of doing business has become a necessity. (Ralph Welborn 2008)

Requirements for interconnection; information risks

Confidentiality: the information sent through that interconnection needs to stay confidential during transit.
Integrity: the integrity of the exchanged information can not be compromised.
Availability: the interconnection needs to stay available.
(James Joshi 2001)

Collaboration is such a way of business: by working together to achieve a common or shared goal, one will be capable of focussing more on his core business. This will allow sharing knowledge and gaining economic benefits in terms of scale and efficiency.
There are many ways to collaborate; from having a joint venture to exchanging goods on the marketplace. Depending on the risks and the reward involved, organisations will have to choose different ways of working together. The parties involved can focus on their own core competence and use each other's knowledge, in order to lower costs of innovation and get a better grip on the market (See paragraph 2.5 for more details). (Ralph Welborn 2008)

Interconnection is a necessity
To create the right environment for such a venture, interconnection has become extremely important, which already is the case for quite some time.
The need for interconnection has been picked up by many organisations in views on software and organisational architectures. Here are a few important ones to this research project:

- **SOA**: Service Oriented Architecture. Even though the primary ideas behind SOA are based upon the reuse of defined services in the organisation (see paragraph 2.2). It is also used to have a better interconnection with the environment directly around the organisation, simply by allowing some of the services to be publicly available to other organisations (mostly done by using web services).
- **SaaS**: Software as a Service. This was known in the past as Application Service Provider, but became successful under its current name, based on the vision of SOA. The idea is to use software from other parties as a service for your own organisation, i.e. getting a complete customer relation management-service system over the web.[6] (see 2.3 for more details)

[5] Varying from business threats like heavy market competition or running behind by not innovating fast enough to technological threats like digital security problems.

[6] See http://www.reeleezee.com, (accessed March 2008) as a good example.

- **The ideas from the Jericho Forum**: The Jericho Forum[7], a part of the Open Group is an impetus for change. The members of the forum have seen that the corporate perimeters crumble due to business drivers demanding greater connecitivity with business partners, suppliers, customers and workers over the internet. This is what they call "de-perimeterisation". As a result of that, they have noticed that a new way of secure collaboration is necessary and they have published a set of "position papers" on the major aspects (see 2.6 for more details).

Focus: Collaboration Oriented Architectures

In order to be capable of interconnecting according to the ideas of the Jericho Forum and still work together, the Forum introduced the Collaboration Oriented Architectures (COA) by use of a position paper. (Forum 2008e)

The Collaboration Oriented Architectures position paper describes a technology and security framework (the COA framework, which will also be refered to as "framework") for sharing information, irrespective of the location of the data, or (trans-)acting parties. It is based on SOA and combines many of the ideas and plans of the Jericho Forum. (Forum 2008e)(See chapter 0 and Intermezzo 1 for more details about the Collaboration Oriented Architectures and the related framework)

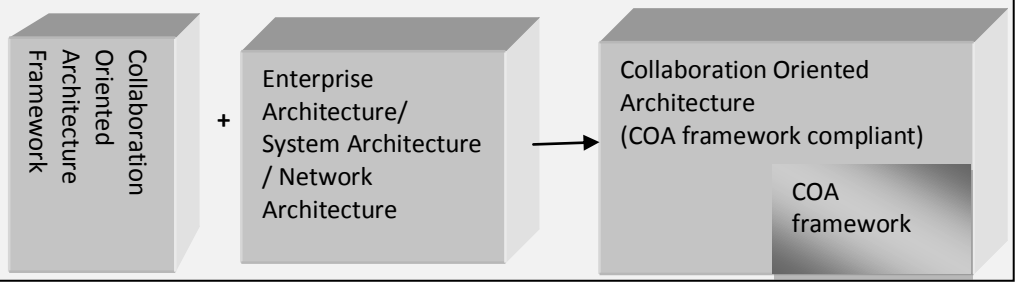
Intermezzo 1: Collaboration Oriented Architectures and the framework in a nutshell

Clearly, the abbreviation 'COA' is used in three varieties. This intermezzo gives a simple view on three COA-related subjects:

- **Collaboration Oriented Architectures (COA)**: Enterprise Architectures that use all the components of the Collaboration Oriented Architectures framework (also refered to as "framework"). This makes the Enterprise Architecture COA compliant. We will use the abbrevation in plural to denote this concept.
- **Collaboration Oriented Architectures framework (COA framework / framework)**: A framework consisting of a set of principles, processes, services and technologies such as:
  - security principles[1]: requirements and constraints like 'know which parties you are transacting with and what level of trust you use'
  - processes: People-, Information-, Device-, lifecycle management
  - services such as: Identity Management and Federation, policy management, information classification, information asset management and Audit
  - technologies such as secure protocols and tools to provide end-point protection that are necessary to collaborate safely.

  The framework includes a set of quality attributes to provide the capability of measuring the success of the framework adoption. (see for the full list paragraph 3.8)
- **Enterprise Architecture / System Architecture / Network Architecture:** these are different architectures that describe an enterprise or a system or a network all of these will be shortly dealt with in this thesis. We will use the abbrevation in singular to denote this concept.

[1]: these elements will be addressed as COA <element>: COA processes, COA services et cetera.



[7] See http://www.jerichoforum.org for access to published position papers.

The framework will be the main focus of this thesis. Multiple items have been written about COA and the related framework. To fully understand what it is about and how it could be used, more study should be done.
Even though there is a variety of elements that can be added to that framework in order to make it workable and complete, this project aims to partially fill the gap of the missing COBIT elements by experimenting with COBIT and COA. We will define and detail the framework based on the current studies and describe a few types of a COA. We will also experiment with one of the COBIT processes with its control objectives to provide extra governance for an Enterprise Architecture class COA.

1.2.Overview

In this subparagraph, some structure will be added to the different elements that are mentioned in the introduction. The elements can be grouped as follows:

- **The basics**: these elements are necessary for today's dynamic corporation. Varying from architecture basics to specialised services, means of interconnection and governance. All of the architectures share these basics and influence the framework. The basics are:
  - *Service Oriented Architecture (SOA)*: SOA will be used both as an architecture principle as well as the basic supplier of many important concepts: security, management, governance, et cetera.
  - *Software as a Service (SaaS)*: SaaS will be used as a special concept of collaboration by means of interconnecting between software services over the internet.
  - *Control Objectives for Information and related Technology (COBIT)*: COBIT will be used as a basic introduction to IT Governance and as an experiment for governing third party services.
  - *Collaboration in theory*: The basic concepts of collaboration as we see on the market today are very important and influential to the COA framework.
- **The COA framework:** the framework as a result of the vision of the Jericho Forum that will transform into the next revolution of secure interconnection and the exchange of information. This should be seen as the centre of this thesis. Based on the Jericho concepts and the influence of the basics, the framework will be further detailed.
- **The Jericho concepts**: the thoughts and ideas of the Jericho Forum will be used as an architecture background vision, as a base for detailing the COA framework and conceptualising the influence of the framework on the existing architectures. It's main theme: the field of de-perimeterisation

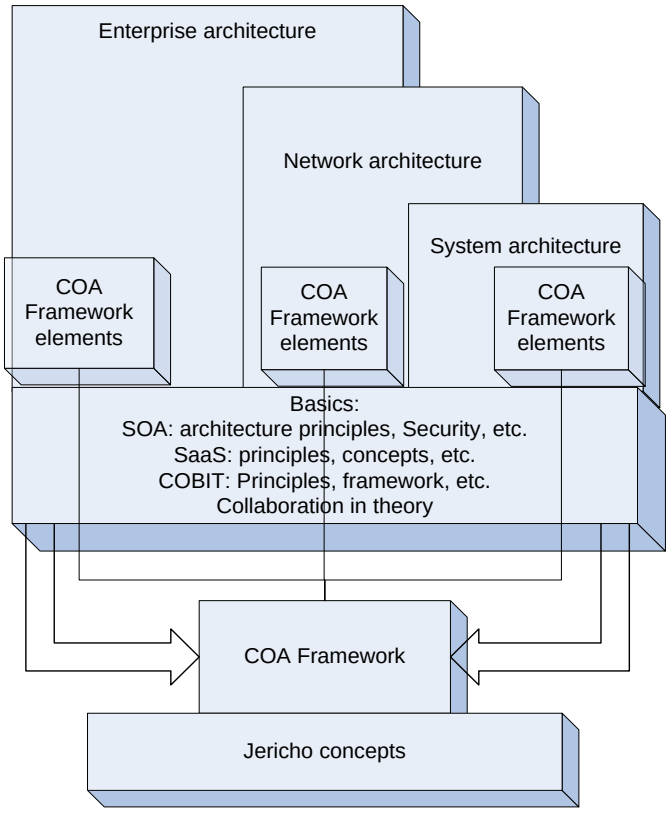


Figure 1: Relationships between the different elements

- **Experimental:** The next and last step will be a hypothetical experiment, grouping some of the framework elements to different example information architectures: the Enterprise architecture, Network architecture and System architecture. These architectures will not be as exhaustively covered as the other elements for the sake of not straying too far from the actual subject. After all, any type of information architecture can adopt the framework elements.

1.3.Research goal

The goal of this research is:

> *"To provide clear and detailed information as to what the Collaboration Oriented Architectures, the encompassing framework and their value is, and how the framework can be adopted, taking into consideration the related concepts of SOA, SaaS, COBIT, the information provided by the Jericho Forum and several important concepts behind collaboration."*

1.4.Problem definition and research questions

1.4.1.Problem definition

As stated in the introduction, the thesis will focus on the Collaboration Oriented Architectures. This provides for the following problem definition:

> *"In order to overcome the challenge of de-perimeterisation, the Jericho Forum launched the concept of COA with the related framework which should be implemented in an information architecture in order to create a Collaboration Oriented Architecture. Until now, there is no such architecture, nor any detailed information of how the framework could be used."*

1.4.2.Research questions

The following research question derives from the problem definition:
"What is the Collaboration Oriented Architectures framework and how can it be used?"

This will be guided by these sub questions:

1. What is the Collaboration Oriented Architectures framework?
2. Why is it important?
3. How can an architecture for a system, for a network and for an enterprise adopt the Collaboration Oriented Architectures framework?

Thus, multiple domains are covered in this thesis. It is not in scope of this thesis to present in detail all the components of any COA. This is why the focus of the thesis will be described per domain in chapter 3.

1.5.Thesis Outline

Chapter 2 provides a clear view on the different domains and subjects used for this thesis. Chapter 3 elaborates on what elements the framework will provide in order to answer question 1 and 2. A more detailed scope of this thesis is given along chapter 2 and 3.
In the concluding chapter 4, all questions will be reviewed once again in order to see what COA and the related framework are about and how they can be used.
Each chapter (excluding chapter 1) starts with an introduction followed by the research work, and ends with a concluding summary.

1.6.Research Plan

Figure 2 shows this research plan consisting of four different steps. It starts with (1) Investigating the elements of the COA framework and the domains coherent to those elements. In order to do so, the ideas of the Jericho Forum are worked through. Other domains are discovered and described as well: A further elaboration on SOA, SaaS, COBIT and Collaboration, using the thesis proposal as a basic point of reference.
From there on, we research COBIT and DS2 (2) to experiment with it. After that, the adoption of the COA framework will be studied (3).We finish by writing the thesis (4).

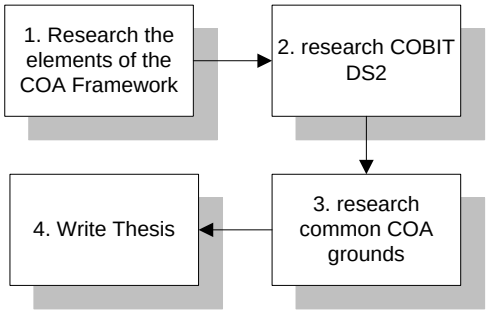


Figure 2: Research path

1.7.Relevance

The research described in this report is relevant in several ways:
First, we gain more insight into COA and the related framework: we come to understand the basic content, the benefits and the use of the content after this study. This allows other objectives, such as:

- The elements of a COA can be further researched in the context of the object where they are integrated into.
- The value of the framework will become more evident, making investments into the framework more interesting.
- The results can be used to create a COA or Jericho licensing program.
- The relations between the different elements of a COA and the framework can be further researched.
- The stage will be set for defining standards for each of the framework elements.
- The experiments of COBIT can be used to further analyse the necessities around governance (or even a new framework) in a COA.

Second, this research allows one to see the framework from multiple perspectives such as systems, networks and enterprise architectures. This also allows us to see whether there are more gaps than already specified in (Forum 2008e).

2. Background

2.1.Introduction

As one might have noticed while reading chapter one: there are multiple domains quite important to this thesis. Each of these domains contain many elements and often synonymous and/or vague definitions. That is why each of the domains must be more clarified in this chapter and a focus will be given for each:

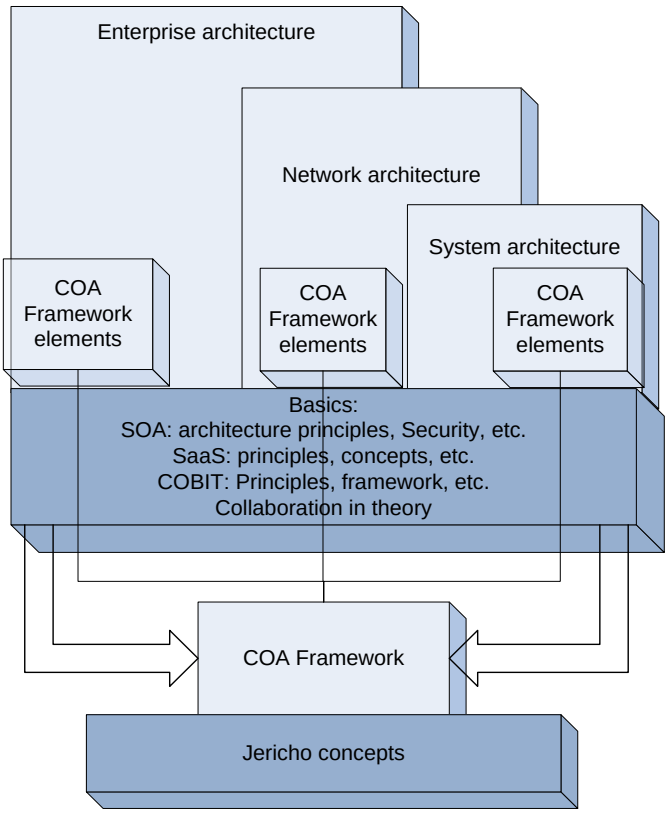


Figure 3: Relationship between the different elements, marked areas will be covered in this chapter.

- In paragraph 2.2 we look at the domain of Service Oriented Architecture (SOA), where a definition will be given based on existing definitions in order to give a better idea of what SOA is. This is very important, because a Collaboration Oriented Architecture framework compliant Enterprise Architecture is a SOA- based architecture with the COA framework implemented. From there on, some important security concepts and implementations will be given on the field of SOA to show what a COA could contain besides the elements of the current framework. At the end of this paragraph, the focus of the thesis will be given for this domain.
- In paragraph 2.3 we look into the domain of Software as a Service (SaaS) from the SOA perspective. A definition of what SaaS is and what issues will rise using a SaaS with a COA follows. At the end of this paragraph, the focus of the thesis will be given for this domain.
- In paragraph 2.4 the Control Objectives for Information and related Technology (COBIT) are described in the context of IT Governance, to show what it is all about. In the same paragraph, other examples of governance- related topics, frameworks and organisations are named. At the end of this paragraph, the focus of the thesis will be given for this domain.
- A small study around some of the basic and most important aspects of collaboration is presented in paragraph 2.5. The results of that are necessary for understanding how to manage and interconnect different Enterprise Architectures that want to be COA framework compliant.
- The concepts of the Jericho Forum are studied in paragraph 2.6 to provide a description of the COA related framework elements to further extent in chapter 3. At the end of this paragraph, the focus of the thesis will be given for this domain: which ideas will be included and which not.
- At the end, in paragraph 2.7, a summary and overview will be given of this chapter.

However, if the reader is already informed on the subjects of this chapter, the author recommends reading only paragraph 2.9.

2.2.Introduction to Service Oriented Architecture

2.2.1.Introduction

The first basic, as seen in Figure 3, is SOA. To be capable of creating a COA, it is necessary to know what the Service Oriented Architecture (SOA) is about: The position paper from the Jericho Forum describes that a COA is a SOA-based enterprise, which is compliant to the COA framework. (Forum 2008e) So what is SOA? And where does it come from?
It is not exactly clear when SOA was introduced. Many sources trace back to Gartner, which, in 1996, was one of the first to describe SOA (Y. Natis 2003).[8]
SOA became a "hot topic" and many concepts were added: Service Oriented Management, Service Oriented Enterprise, Service Oriented Computing, Service Oriented Science. Everything should be "Service Oriented". (Erl 2005)

Besides the many concepts that have been introduced, people started giving their own definition of what Service Orientation should be and thus many ideas arose on what SOA is. This makes it easier to adapt the right SOA strategy for a company (one that fits). (Henry Peyret 2005)
All these definitions are also backed up by groups that tend to say "they have the right standards".[9]

In order to present a proper introduction to SOA, it seems best to first give a good definition of what it really is. This has to be done by searching for definitions from different sources, extract the important elements from them -and their sources- in order to create a definition that contains the most important elements from those definitions. This definition is given in section 2.2.2.
After creating a definition, some important ideas and constructions based on SOA are analysed, to get a more detailed grasp of what an implementation of SOA could include. Furthermore, we will discuss the following:

- In section 0 we have a look at SOA and its security. The COA framework is a security framework. Therefore, to understand what the COA framework delivers in terms of value and security, we need to understand what the SOA security provides.
- In section 2.2.4 other concepts are introduced, to see which ones can be found in a SOA-based environment or enterprise.
- In section 2.2.5 an explanation follows of why using SOA as a base for COA is quite important.
- In section 2.2.6 a concluding summary will be given containing the information in this paragraph.

2.2.2.Service Oriented Architecture

A Service Oriented Architecture (SOA) is an architecture based on the use of modules called services. These services provide a way to shape both Business and Technology. (Alan 2002; David Sprott 2003; Massuthe 2005; Rogers 2005; Hurwitz 2006; Mike Craig 2007; Nickul 2007)
SOA is based on the following elements:

[8] http://www.rgoarchitects.com/Files/SOADefined.pdf,

[9] A few groups and URLs linking to them: Open SOA Collaboration: http://www.osoa.org/display/Main/Home, the SOA Working Group from the Open Group: http://www.opengroup.org/projects/soa/, the SOA Consortium: http://www.soa-consortium.org/, OASIS SOA committee: http://www.oasis-open.org/committees/tc_cat.php?cat=soa , the CBDI Service Oriented Practice Portal: http://www.cbdiforum.com/

Service brokers?

A service broker might sound strange to many of us. There are however different examples of a service broker: from a repository with a database, to the well known "service bus" or "enterprise service bus".

- **Services** that represent either business or IT functionalities/tasks, that are:
  - Reusable (by other services) (Y. Natis 2003; Rogers 2005; Hurwitz 2006; Natis 2006; Poppensieker 2006; Surekha 2006).
  - Loosely-coupled (Natis 2006; Poppensieker 2006) to ease change (Gilpin 2005).
  - Relevantly granular to the service requester (David Sprott 2003).
  - Providing a neutral/standard interface that is discoverable. (Alan 2002; David Sprott 2003; Y. Natis 2003; Giudice 2004; W3C 2004; Heffner 2005b; Krafzik 2005; Massuthe 2005; Rogers 2005; Trieloff 2005; Poppensieker 2006; Services 2007; Foster 2008)
  - Independent(-ly designed) as a module. (Rogers 2005; Trieloff 2005; Hurwitz 2006; Services 2007)
  - Able to hide certain implementation details (David Sprott 2003) or even independent of its implementation (Hurwitz 2006; Services 2007).
- **Service providers**, that provide services (Massuthe 2005), the services that they provide will be described by certain semantics and sometimes even give details on the implementation of the service, to see if it is tailored to the needs of the consumer (C. Matthew MacKenzie 2006).
- **Service requester** (David Sprott 2003; Massuthe 2005) **or Service consumer**, that requests and in a next stage consumes a service, when the service provider allows it. (Y. Natis 2003; Jørstad 2005; Rogers 2005) They may or may not be able to see the exact provider, yet can find it via the service broker (C. Matthew MacKenzie 2006).
- **Service broker**, that connects a group of service providers and requesters/consumers through allowing the service requester to register himself to a broker by submitting information about how to interact with its service. (Massuthe 2005)

The actual deployment and the precise form of a SOA are not included in the definition, because it may vary for each organisation and environment. In that way extra flexibility is added, which makes it an even more powerful architecture . (Surekha 2006; Earl 2007)
We conclude the definition of SOA with an important quote that highlights the most important aspects of SOA:

> *"SOA is a means of organizing solutions that promotes reuse, growth and interoperability. It is not itself a solution to domain problems but rather an organizing and delivery paradigm that enables one to get more value from use both of capabilities which are locally "owned" and those under the control of others. It also enables one to express solutions in a way that makes it easier to modify or evolve the identified solution or to try alternate solutions. SOA does not provide any domain elements of a solution that do not exist without SOA."*
>
> (C. Matthew MacKenzie 2006)

However, SOA is a way of organising solutions that promotes reuse. One could argue that it may become unusable in environments and cases where reuse or reorganisation of the current solutions is not involved. In fact there is a growing group of people which argues that that the services in SOA are a worthy continuing way to go, but the architecture bit in SOA only adds complexity, confused thinking and no real effective value to their opinion.[10]

2.2.3. SOA and Security

In this section, we have a look on SOA and Security. Based on the definition from the previous section, there should be some security measures incorporated in business, as well as in IT[11].

[10] See http://broadcast.oreilly.com/2009/01/soa-is-dead-its-about-time.html and http://apsblog.burtongroup.com/2009/01/soa-is-dead-long-live-services.html visited at 14-02-2009

[11] Some of the IT-security measures are be described here, others, more generic, will be described in section 2.6.4.

Both will be discussed in this section. Starting here, we continue to monitor some of the security issues and end the section with an overview of where the security measures in a SOA can be found. It is not a purpose here to list all security details and locations of the measures. Only some of the important ones will suffice.
It is elemental to remember that the taken measures will be implementation-specific. In fact, some of the implementations neglect Business Security as far as being a part of the SOA security is concerned. They focus solely on technology and thus the technology-based security measures.

Business Security
The Business security recommendations that have been found so far are implementation dependable. Should a strict software approach be chosen for instance, many of the following recommendations can be ignored. However, in the following we use the approach of an implementation that is focussed on both IT and business:

- **Vision, architecture and strategy:** SOA will influence your vision, architecture and even your strategy. Failing to recognise so may lead to creating certain problems. That is why the following recommendations have been made in related literature:
  - *Setting up a SOA-Vision:* A vision for the impact and structure of SOA is always necessary to understand the impact on both Business and IT for breadth, as well as depth of the effects. If one does not understand the total impact and misses out important components, a faulty implementation of SOA may lead to extreme vulnerability. An important part of the SOA-vision is a security vision that shows the exact security measures needed, and how they should be implemented. Once the SOA-Vision is established, the designing of interfaces and services commences.
  - *Creating a SOA roadmap:* Create a roadmap of what elements to build in early stages and which ones in a later phase. Make sure all the required components for the implementation are there and that it is feasible to do, taking into consideration the position of the enterprise at that moment. A basic element of this roadmap is security planning, in which all the security aspects should be named. This roadmap can be top down (starting with the SOA vision) or more pragmatic (starting with the data). The first approach is semantically richer and often more secure, the second is rather reliable and often easily executed.
  - *Governance processes on architecture and Service (Interface) design and implementation:* The Service Interface Designs form the digital model of the business. It is important to create governance structures for developing good Service Interface Designs. They must be both practical to implement and give a wide business insight. This also counts for the services itself: one should have governance processes to see the service lifecycle being fulfilled. There is also a good governance process necessary to make sure that the complete architecture will be secure and fulfil the business needs. All governance processes should be end-to-end in both the architecture depth/width and the lifecycles. The latter is extremely important, since SOA will be quite the dynamic type of architecture with many evolvements of services.
  - *A Service Oriented Security Architecture (SOSA):* This is an architecture that comprises all of the services in a conceptual, logical and physical description. It may also contain a business interaction model that describes the interaction between services and humans. It shows where certain security measures should be and how they should be implemented. There are different possible implementations to these architectures, such as the by OASIS defined Web Services Security Model, or the SOS-model by the Arctec group.
  - *Setting up a SOA strategy that takes all this in mind:* One should have a SOA strategy that starts small and is engineered in such a way that all of the aforementioned recommendations are implemented. It should be set up systematically without forgetting any of the recommendations. This ensures all of the (business and) security requirements being in place.

- *A SOA Funding model:* One has to make sure that there is a good funding model allowing a SOA to gradually mature and be the fulfilment of the original SOA vision. Obviously, nobody wants (security) gaps as a result of running out of money.

- **Systems, services and processes:** Having a strategy within SOA does not mean all aspects are covered. In order to get the full promised benefits from SOA and still work securely, one will have to consider these recommendations:
  - *Avoid binding to a single technology or process type:* Avoid binding to a rigidly constant set of behaviours and/or interfaces. This will compromise cooperation with second and/or third parties. Whenever one has adopted rigid systems or processes, loads of conversions and often insecure mechanisms have to be used to exchange information between the rigid systems and other parties.
  - *Reusing legacy systems safely:* Many companies are attracted to SOA because of the legacy leverage. These legacy applications can be unsafe and create a wide variety of problems. Herein lays the need to identify and mine the services from existing systems, to see whether they are safe to use, or encapsulate them in such a way that they can be made safe. Yet, it is important to keep business rules and decision-making logic outside of the legacy system and inside something like a business rule management system.
  - *The Service Delivery Life Cycle:* There should be a good service delivery lifecycle with all the governance processes in place. The lifecycle should be managed and effectively monitored. It is often also called the SOA lifecycle. What the lifecycle will eventually look like is implementation specific. As an example: *"Assemble, deploy, manage and model"* (Rob High 2005). The cycle does not have to flow linear, as long as it will flow systematically and well governed. Because of many dynamic organisations wanting to be even more dynamic by using SOA, the service delivery life cycle will flow even faster. That is where good governance once again comes into play.

- **Others:** The following recommendations have been found in the literature around the usage of SOA:
  - *Training and the use of best practices:* In order to fulfil the SOA strategy, different units inside the enterprise have to be trained in order to deliver high quality and good security via the best practices. Even though this is not new (other types of architectures will need this as well), it remains elemental since SOA results in a whole new dynamic process.
  - *Think about collaboration:* It is important to think about collaborations and partner services: will there be any? Moreover, if so, can they be incorporated into the system securely? Make sure there is a section (or multiple sections) in the architecture that ensures secure interoperability with other types of SOA.

(Heffner 2005a; Peterson 2005; Pezzini 2005; Rob High 2005; Sluiter 2006; Hutinski 2007b; Amelia Maurizio 2008; Liam O'Brien 2008)

There are many other recommendations to consider. Some of them may be noticed in section 2.2.4.
Many of the listed recommendations here are actually already familiar. Most of them have existed for long, only in other types of architectural concepts.
(Hutinski 2007b)

Technology Security
Different sources show different security measures, depending on the type of implementation that is used. In this subsection, we explore which security protocols can be used when web services are used in its SOA and are XML based. (the list is not exhaustive, see (Jothy Rosenberg 2004) for more details)[12]

[12] Another couple of sources where every detail of these standards can be found: www.w3.org and www.oasis-open.org , both visited at 8-08-08

In order to give some structure to the list, we use the WS-Security standard and the Web Service Security stack as a point of departure. We subsequently handle most of the security standards in a bottom-up order, based on the web service security stack, finishing with some not to be overlooked XML-related security mechanisms.
This is done based on the Web Service Security stack as seen in Figure 4.

Technology Security-Web service security:
The following web service security related standards are considered very important:

- WS-Security (Web Services Security):
  - *What is it?* - It is an open OASIS standard[13], focused on SOAP security: securing the content of Web service messages either in use, transit and/or storage. It is primarily a security layer that focuses on applying existing security technologies such as X.509 certificates, SAML assertions, XML Signature, and XML Encryption to SOAP messages. It does not create a secure pipe like SSL or IPSec(see section 2.6.4), yet it secures the messages themselves by the use of the mentioned technologies.[14]
  - *What does it contain?* – The standard contains several elements:
    - A definition for a WS-Security header that provides a standard place to put security tokens. Other elements from protocols such as XML Encryption and XML Signatures can be set there as well.

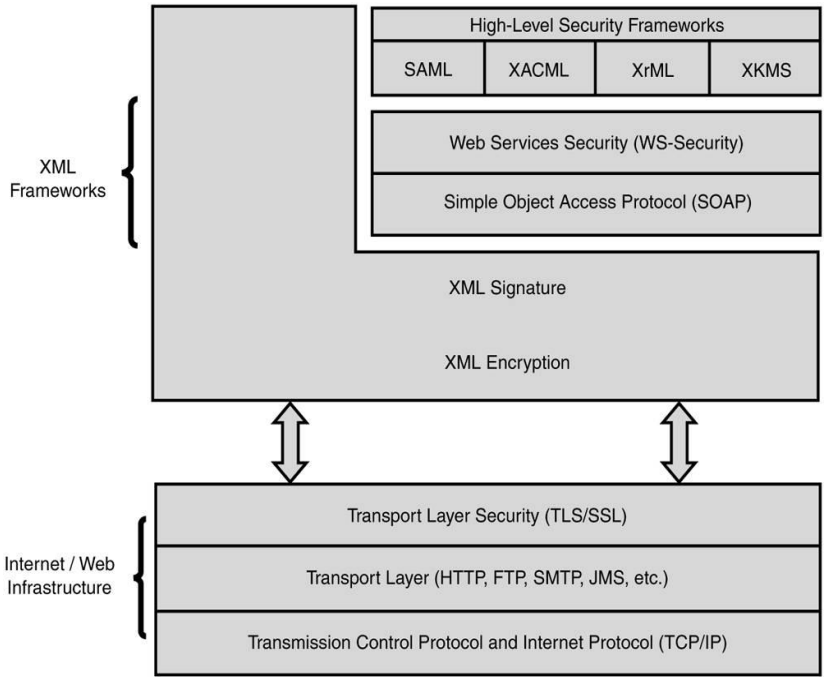


Figure 4: The Web Services Security Stack in context (Jothy Rosenberg 2004)

    - A WS-Routing protocol that is a stateless protocol for exchanging one-way SOAP messages from an initial sender to the ultimate receiver, potentially via a set of intermediaries.
    - A set of extra rules and features for XML Encryption within SOAP messages.
    - A structure for setting a message timestamp in the WS-Security header.

[13] See http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss and www.oasis-open.org for more details. Visited on 09-08-08

[14] The WS-Security holds the Web Services Security Stack and is therefore the first standard to be described in this list.

- *Why is it important?* - It is a very important standard for securing web services. Many companies like Microsoft and IBM have strongly committed to it. It also gives a good alternative for secure pipe protocols (see (Jothy Rosenberg 2004) for a good comparison). IT is also a critical standard to other security items, such as WS-SecureConversation, WS-federation, WS-Authorization, WS-Policy, WS-Trust, WS-Privacy and many other higher layers (see also Figure 4). Using this standard, should make it easier to use the other already named web service security protocols and standards.

- XML Encryption:
  - *What is it?* – It is an XML based W3C recommendation[15], which is focused on encrypting an arbitrarily sized XML message in an efficient way. It is based on shared key encryption. It provides message confidentiality (see also section 2.6.4). It is commonly used when shared services are utilized or when the message is stored (Even after it reaches its destination). One can either encrypt parts of a message or the complete message. It is a W3C recommendation that consists of the syntax and the mechanisms that the protocol should provide.
  - *What does it contain?* - XML encryption consists of EncryptedData, which holds the encrypted data, the EncryptedKey that is the session key for exchanging the shared key, the KeyInfo gives the actual key, a hint or pointer to the key, the CipherReference: a pointer to an external source that is encrypted. EncryptionMethod is a pointer to the algorithm and CipherData/CipherValue/CipherReference URI are holding (pointer)data to encrypt. There are obviously other objects in XML encryption and different methodologies of use, yet they are irrelevant to the thesis.
  - *Why is it important?* - It is a core foundation of the WS-Security standard because it handles the confidentiality issues around XML. It is an important basis to many other protocols such as XKMS, XACML and SAML (see for more details around these protocols further ahead in this section). Combining it with XML Signature makes it a really firm security protocol.
- XML Signature:
  - *What is it?* - It is an XML based syntax description, a W3C recommendation[16]. It is designed to provide a mechanism for identity, message integrity and non-repudiation (see also section 2.6.4). It allows one to encode digital signatures into XML. One can either sign different parts of a message (such as XML or an image) or the complete message. It is a W3C recommendation (in cooperation with IETF) that consists of the syntax and the mechanisms that the protocol should provide.
  - *What does it contain?* - An XML Signature itself consists of a set of pointers (URL: Uniform Resource Locator) to things to be signed, the signature, an optional key for verifying the signature and an optional Object tag that has anything else with it. There are different types of XML signatures and different procedures for signature and validation in the standard. Detailed descriptions hereof are beyond the scope of the thesis.
  - *Why is it important?* – By providing a mechanism for message integrity and non-repudiation, it became a core foundation of the WS-Security (Web Service-Security) standard. It is an important basis to many other protocols, such as XKMS, XACML and SAML (see for more details around these protocols further ahead in this section). Combining it with XML-encryption makes it a good security protocol.
- SOAP (Simple Object Access Protocol[17]):
  - *What is it?* - It is an XML transport protocol specification that became a recommendation of W3C. SOAP defines how the data moves from one place to another over a network.

[15] See also http://www.w3.org/TR/xmlenc-core/ for more details, visited at 09-08-08.

[16] See http://xml.coverpages.org/xmlSig.html for more details as well, visited at 09-08-08.

[17] The Simple Objects themselves and the related

SOAP allows the sender and receiver of XML documents to support a common data transfer protocol.
  - *What does it contain?* – The artefact described by the protocol (a SOAP message) consists of a SOAP Envelope that contains a SOAP Header and a SOAP Body. The envelope creates namespaces and indicates the start and end of the message. The header contains directive information and the body contains the payload.
  - *Why is it important?* - The header is important for security (such as the WS-Security header). Furthermore, SOAP is the standard for transporting XML, which automatically defines it as an interesting object in terms of security.
- SAML (Security Assertion Mark-up Language):
  - *What is it?* SAML 1.1 is an OASIS-approved standard. It allows the description of security assertions in XML. A security or trust assertion is defined in SAML as a claim, statement, or declaration of fact according to some assertion issuer (SAML authority), specifying Authentication, Authorization[18] and attributes (that provide qualifying information about either an authentication or authorization assertion). By using SAML, one can describe and use security assertions, possessing a form of portable trust among multiple trusted domains (like multiple companies) which allows the usage of a Federated identity. See for more details and newer versions section 2.6.7.
  - *What does it contain?* – It has three XML-based mechanisms:
    - Assertions: An XML Schema and definition for security assertions. This makes SAML an XML framework that can be extended with new assertions.
    - Protocol: An XML schema and definition for a request/response protocol. The requests are there for policy decisions and enforcement from SAML authorities.
    - Bindings: Rules on using assertions with standard transport and messaging frameworks. These rules are described as a set of bindings and protocols.
    - See also (Barannikov 2008) for more details and a good comparison to WS-Federation.
  - *Why is it important?* – SAML addresses the problems around portable trust. This sets a technologic basis, allowing federated identity with the use of web services. It is a core language for WS-Security and often used as the language for portable trust. It may be considered quite complex.
- XML Key Management Specification (XKMS):
  - *What is it?* - It is a W3C recommendation[19]. It specifies protocols for distributing and registering public keys (for a PKI system) suitable for use in conjunction with the XML Digital Signature standard and the XML Encryption standard. It has two goals:
    - To support a simple client's capability to use sophisticated key management functionality. Such a simple client is not concerned with the details of the infrastructure required to support the public key management, but may choose to work with X.509 certificates if it is able to manage the details.
    - To provide public key management support to XML applications.
    - It needs a request/response mechanism to use PKI.
  - *What does it contain?*- IT is composed in two parts:
    - XML Key Information Service Specifications (X-KISS) is a protocol to support the creation of a service to which an application delegates the processing of Key Information.
    - XML Key Registration Service Specification (X-KRSS) is a protocol to support the registration and management of a key pair by way of a key pair holder, with the intent that the key pair is subsequently usable in conjunction with the XML Key Information Service Specification or a Public Key Infrastructure such as X.509 or PKIX.
    - The exact details of these parts are deliberately out of scope in the thesis.
  - *Why is it important?* – Web services need end-to-end message integrity and confidentiality, which means that they need XML Digital Signature and XML Encryption.

[18] See also section 2.6.4for an explanation around Authentication and Authorization.

[19] See also: http://www.w3.org/2001/XKMS/, visited on 9-08-08

Those technologies, in turn, scale best when they use public key cryptography. Public key cryptography needs a supporting infrastructure, PKI, to handle distribution, certification, and life-cycle management (for example, revocation) of keys. Web services themselves provide a powerful new approach to PKI that prevents each Web service requester and provider from having to build their own PKI: accessing a trusted PKI as a service. XKMS aims to do just that.

- eXtensible Access Control Markup Language (XACML):
  - *What is it?* – It is an XML schema for representing authorization and entitlement policies. It is an OASIS Open Standard, thus representing the rules that specify the who, what, when, and how of information access. Access control determines who can look at something, what they can do with it, the type of device they can see it with, and so on.
  - *What does it contain?* – it has a set of objects and methods to control access policies. It uses Policy and Policy Set as authorization policies. Each policy document contains one Policy or Policy Set, a policy is expressed through a set of Rules. A Policy represents a single access-control policy, expressed through a set of Rules. A Policy is intended to form the basis of an authorization decision. A PolicySet contains a set of Policy or other PolicySet elements and a specified procedure for combining the results of their evaluation. This is the standard way of combining separate policies. A Rule contains a Boolean expression that can be evaluated in isolation as the basic unit of management; it can be reused in multiple policies. A Target defines a set of resources, subjects, and actions to which a certain Rule is intended to apply. It is the set of decision requests that a Rule, Policy, or PolicySet is intended to evaluate. An Obligation is an operation specified in a Policy or PolicySet that should be performed in conjunction with the enforcement of an authorization decision. A Condition is an expression that evaluates to either True, False or Indeterminate. The Effect is the intended consequence of a satisfied Rule — either Permit or Deny.
  - *Why is it important?* - Even though XACML is quite complicated, it still is a very important standard because of the necessary Access Control. XACML is elemental because:
    - Computing systems are extremely general. They have the broadest possible set of privileges of accessing data and applications. Inherently, they can also access those systems with little or no security policies, making themselves very insecure.
    - Access control policy enforcement is handled at many different points. In cases of reasonably strict security policies, systems are access-controlled at the point of deployment. Enterprise security policy has many elements and points of enforcement, including HR, Finance, Legal, and others.
    - There are different access control enforcement mechanisms. Each point of enforcement is typically managed independently to make sure the policy is implemented accurately. This makes it prohibitively expensive to modify security policy. It is impossible to obtain a consolidated view of the overall security situation in an enterprise.
    - The number of reasons will grow with the oncoming machine-to-machine interactions of Web services. They will exacerbate these issues.[20]
- eXtensible Rights Mark-up Language (XrML):
  - *What is it?* – it is a rights language that supports a wide variety of business models, from free content where still needs to be controlled who accesses it (for example, real estate home listings), to valuable content that is purchased by the end user (for example, digital music). It can specify simple and complex rights and is designed to handle any type of digital content or service. It gives precise meaning to all components of the system. A couple of its critical early design goals were that it should be interoperable with other standards and specifications, and it should be platform neutral.

[20] XACML is also important for various other reasons, see paragraph 2.6 for instance: XACML can be used for the Jericho Policy Management.

- *What does it contain?* It contains a Data Model (and a few other definitions out of the scope of this thesis). This data model consists of four entities and the relationship between them. The most important relationship is the XrML assertion Grant, which consists of:
  - The Principal to whom the Grant is issued.
  - The Right the Grant specifies.
  - The Resource that is the direct object of the "rights" verb.
  - The Condition that must be met for the right to be exercised.
  - The four objects are:
    - A Principal is an individual who must present identification credentials such as an X.509 certificate or a digital signature.
    - The Right is a verb that can be granted to a Principal to exercise against certain content.
    - The Resource is the object to which a Principal can be granted a Right.
    - A Condition specifies the terms, conditions, and obligations under which the Right is exercised.
- *Why is it important?* – Even though XACML can actually fulfil the tasks of XrML, it is a good effort to handle digital rights management and to let different types of digital right management systems work together.

- WS-Policy:
  - *What is it?* It is a framework that provides mechanisms for exchanging requirements between web service providers and requesters[21]. It can be used on top of the Web Service Security stack. WS-Policy is actually a set of specifications providing generalised mechanisms for describing policy, in a machine-readable way. The idea is there ought to be a common language for describing the rules to interaction with a Web service, or what a client requires of a Web service, regardless of whether the domain is security, privacy, transactions, or any other category. In addition, some of these rules, called assertions in WS-Policy, are common across all domains. An artefact based on these rules (a WS-policy) can be a standalone object that points to certain WSDL files, or certain web services. It could also be pointed at by a part of a WSDL file or by a web service.
  - *What does it contain?* – The WS-Policy Framework contains a set of specifications relating to policy:
    - WS-Policy defines a framework for describing policy assertions. It is a generalised grammar for describing the capabilities, requirements, and characteristics of a web service. A policy assertion describes requirements that a web service or client must adhere to. For example, the server might specify an encryption algorithm to be used when encrypting messages bound for the service. WS-policies are wrapped together by a Policy element.
    - WS-Policy Attachment describes how these policies are attached to a resource. This can be done by either defining the policy within the definition of the web service (in WSDL for instance), or by creating a standalone policy that points to the web service.
    - WS-PolicyAssertions describe a common set of assertions that are applicable across different domains: the character-sets that are supported, the supported languages, some version-management, and specific grammar-related assertions.
  - *Why is it important?* - It is important for multiple reasons:
    - It allows developers to express requirements clear and complete, more than with WSDL.
    - The ability to specify what exactly should be signed or encrypted in a message.
    - The ability to specify the messages and the kind of security necessary to use the service.
    - WS- Policy sets the stage for WS-SecurityPolicy, which allows defining security policies based on WS-Policy (WS-SecurityPolicy is outside the scope of the thesis).

[21] See for the latest W3C recommended version : http://www.w3.org/TR/ , visited at 09-08-08.

- **WS-Privacy:**
  - *What is it?* - It is a proposed standard that uses a combination of WS-Policy, WS-Security, and WS-Trust to communicate privacy policies. It can be used on top of the Web Service Security stack. It is designed for use by organisations that deploy web services and require that incoming SOAP requests contain claims where the sender conforms to the service provider's privacy policies. WS-Security encapsulates these claims into security tokens that are verified before accepting any incoming SOAP requests. It looks like W3C's Platform for Project Privacy Preferences.
  - *Why is it important?* – WS-Privacy can help a user expressing privacy preferences and notify the user when something occurs in conflict with the privacy preferences. It can also be used to let services decide whether some personal information is revealed or not.
- **WS-SecureConversation:**
  - *What is it?* WS-SecureConversation 1.3 is an OASIS standard[22]. WS-SecureConversation establishes a mutually authenticated security context in which a series of messages are exchanged. It is an optimisation for WS-Security to use multiple messages. WS-SecureConversation uses public key (asymmetric) encryption to establish a shared secret key and from there on uses shared key (symmetric) encryption for efficiency. The same shared key is used to encrypt a series of SOAP messages.
  - *What does it contain?* – The standard describes a SecurityContextToken-tag that is a security token, which provides the base for setting up the Secure Conversation. Furthermore, this tag has certain elements and attributes which allow establishment of a shred secret security context in WS-security. The exact details are however beyond the scope of the thesis.
  - *Why is it important?* – This security standard provides a more efficient alternative of message transport to the WS-security based point-to-point transport if massive amounts of messages have to be encrypted.
- **WS-trust:**
  - *What is it?* – WS-Trust is an OASIS open standard[23] that defines extensions to WS-Security, which provide methods for issuing and exchanging security tokens. It creates ways to establish and access the presence of trust relationships. Like SAML, it defines a request/response mechanism for obtaining security tokens. It can use WS-Security, WS-Policy and a few others as building blocks. In this case, even Web services themselves may be a security token. It can be used on top of the Web Service Security stack.
  - *What does it contain?* The WS-Trust standard entails a WS-Trust Model, a few methods and processes to exchange the security tokens and to request them. Furthermore, it allows a requester to demonstrate its ability to prove a required set of claims. The basic idea behind the model and its processes is that it allows web services (and by that the user of those services) to trust each other. The details of WS-Security and WS-trust are not included in this thesis.
  - *Why is it important?* - The matter of trust is important to Web-services, especially when one works with different organisations and thus different trust domains. This standard assists in working efficiently and in a trustworthy way. Yet, it could fail if the model is implemented faulty. WS-Trust is also used in other WS-standards such as WS-federation.
- **WS-Federation:**
  - *What is it?* - It is a standard used by IBM, Microsoft among others. It still seems to be in development within OASIS. It describes how to manage and broker the trust relationships in a heterogeneous federated environment, including support for federated identities. Federation in this context means a group of organisations that have communicating Web services, agree on a uniform set of standards and policies about identification and

[22] See also docs.oasis-open.org/**ws**-sx/**ws-secureconversation**/200512/**ws-secureconversation**-1.3-os.html for more details. Visited on 09-08-08.

[23] See also http://docs.oasis-open.org/ws-sx/ws-trust/200512 for more details, visited at: 09-08-08.

authentication of entities. The purpose is to translate one entity's security tokens into another type of security token. It uses WS-Trust, WS-Security, WS-Policy and WS-Secure Conversation as building blocks.
  - *What does it contain?* – It contains a model and a description of the process to establish the trust between different trust domains. However, the details and the other content of this standard are not the scope of this thesis. See also (Barannikov 2008) for more details and a good comparison to SAML.
  - *Why is it important?* - See SAML for the reasons of its importance. It is not as much accepted as SAML. See also (Barannikov 2008) for more details.
- **WS-Authorization**
  - *What is it?* - It is an OASIS specification[24]. It deals with authorization decisions in the context of Web services and describes how access policies for a Web service are specified and managed. As of this writing, no specification has been published for WS-Authorization. Its objectives are similar to the eXtensible Access Control Mark-up Language (XACML) and are undoubtedly heavily influenced by it.
  - *Why is it important?* – Authorization is an important aspect of Web-Services and of security in general.

Technology Security – basic security measures for XML:
The following standards are basics to XML security:

- XML-Schema (extensible Mark-up Language)-Schema:
  - *What is it?* - It is an XML-based W3C recommendation[25] that uses XML-based artefacts that can be used to define and validate an XML document. The artefact can exist, independent of the XML-document that has to be defined/validated. It can be used to supply constrains for minimal and maximal values (like maxima and minima in values such as temperature).
  - *What does it contain?* - The artefact described by the protocol (an XML Schema) contains constraints for values and definitions for the XML document lay-out.
  - *Why is it important?* – This Schema is important to ensure no invalid values are used, it is also a basic component of WSDL.
- WSDL (Web Service Description Language):
  - *What is it?* - An XML-based (language) protocol or XML format which defines the input and output structures for Web services. It also defines the payload one can find in a SOAP message. It does so by creating a schema for XML/SOAP objects and interfaces.
  - *What does it contain?* – A set of rules and tags are described: A definitions tag that establishes the namespace, the operations a service or SOAP-object can perform, defined by message and port type. How the service can be invoked defined by binding tags, and the location of the service within the service tag.
  - *Why is it important?* - It defines the structure of SOAP messages as well as the services. Many (development) tools to create, update and use the services and send and receive the correct SOAP messages, often use the artefacts. They is a huge security threat if the WSDL-files are just published online. Everybody will be able to see what the services and messages do, how they work and where they are to be found.

Technology Security – Others:
There are also other security measures inside SOA, they are either described in paragraph 2.6 or in chapter 3, or deliberately out of scope in the thesis.

[24] See also http://www.oasis-open.org/committees/wsfed/ for more details, visited at: 09-08-08.

[25] See http://www.w3.org/XML/Schema for more details as well, visited at 09-08-08.

Some security issues in SOA:
There are still many security issues in a SOA that seem hard to address. It is good to observe some of them, in order to see that SOA security is not perfect. Even though the details are out of the scope of the thesis:

- One of the security issues in SOA lies in the trend of making services that are too general. Since they are broadly oriented, they can often communicate with any kind of other service. This makes it hard to track a service in communication with the right parties and allows it to actually exchange information with an uninvited party. (Hutinski 2007b)
- For the same reasons as mentioned above, data security and data control became a security issue as well. (Hutinski 2007a)
- Another security issue is the lack of good governance, such as SOA Governance (See next section). Many companies subdued to a faulty implementation, creating a load of troubles on the field of service integration and the service lifecycle (Hutinski 2007b).
- The WSDL files may also become a huge security problem, just like UDDI and other publications about services. If one publishes them in the wrong place without the proper protection new security threats arise. (Jothy Rosenberg 2004)
- The greatest challenge in implementing and establishing SOA is prevent a lack of acceptance and a lack of understanding it, which will create many problems on different fields, such as security and quality. (Ohrstrom 2007)
- One still creates security issues by improper selection of application infrastructure components, insufficient validation of the SOA enabling infrastructure implementation, SOAM, SOI and SOE. (Liam O'Brien 2008)
- Multiple security frameworks (Ivar Jørstad 2005; Bingnan Xiao 2006a; Ismail Khriss 2007; W.T.T sai 2007) and measures (X. Zhou 2006; Michael Menzel 2007; W. T. Tsai 2007; W.T.T sai 2007; Surya Nepal 2008) arose to handle dynamic (and/or flexible) collaborations[26], often between multiple parties[27], yet none of them seems to become the workable standard and still a good solution is missing.

There is obviously a wide variety of possible obstructions, of which most are implementation-specific. See also (Mamoon Yunus 2005; Yuri Demchenko 2005) for a list of SOA related attacks, exploits and threats (that are implementation specific)

Where are the IT-measures then?
So, where could one find the IT-measures in a SOA? One can enable security measures in many places of the SOA-components, like:

- **In specific hardware:** There are special security devices, such as:
  - *XML gateways* or *XML firewalls*: from different manufacturers like: Cisco[28], Vordel [29] or Sun[30]. The XML Firewall[31] operates as a stateful firewall, yet focuses on XML. It provides the functions:
    - XML threat protection: Attackers can use different attacks like Denial of Service, on XML processors, by sending wrongly formatted messages. These are filtered by an XML firewall.

[26] These collaborations can vary between sharing resources to sharing complete services and/or processes.

[27] These parties vary from other services to physical other parties with other security domains.

[28] See http://www.cisco.com/en/US/products/ps7314/index.html for more details. Visited on 09-08-08.

[29] See http://www.vordel.com/products/vx_gateway/ for more details. Visited on 09-08-08.

[30] See http://www-306.ibm.com/software/integration/datapower/ for more details. Visited on 09-08-08.

[31] The XML firewall is often also called an XML gateway and vice a versa. The strict difference between a gateway and a firewall has decreased by many overlapping functions and technologies.

- Schema validation (incoming): Every incoming message is validated by the use of XML-schema (or XSLT) and various WS standards. All invalid ones are dropped. This is to make sure that none of the SOA services will fail based on wrongly formatted messages.
- Schema Validation (outgoing): Every outgoing message is validated as well, this is to make sure that all services behave correctly. If not, an attacker might have been successful and an alert should be sent.
- Integrity Protection and authenticity checks: Ensures that a third party has not modified an incoming message. Digital signatures and other WS mechanisms can be used and checked.
- XML encryption services, like the XML security appliance from Forum Systems, encrypt XML in real time as the data approaches the servers, it decrypts the data as it exits the server. It examines data on a tag-by-tag basis and does not encrypt unnecessary or non-critical fields
- Other optional security services: XML firewalls can, depending on the implementation, be enriched with multiple checks by installing processing entities that check the message when the firewall service has obtained it. This can vary from anti SQL-injection services, IP-checking services, to content-based routing, service negotiation and QoS.
- An XML firewall can also be a piece of software running on any kind of platform. It is very platform specific as to how it exactly works, though. Most of the time it follows the usual concept of a firewall: a processor accepts a message and forwards it to agents, to check the message (see also section 2.6.4)[32]. An XML gateway is sometimes called an XML firewall. See also for more details: (Heasley 2005; Mamoon Yunus 2005; Yuri Demchenko 2005; Sam Weber 2007)

- *XML accelerator*: XML accelerators are hardware-based XML processors, they will speed up XML processing and are often equipped with one or more security measures.[33]
- *Integrators*: Integrators convert XML messages to other formats and types of messages or even to programming code and vice a versa. The integrators often have one or more security measures installed for a secure conversion.[34]
- *Application Servers (AS)*: These work differently: They are responsible for executing applications. AS can be organised in a hierarchic tree structure, where an application is executed in its branches[35]. Whenever a service is running on an Application Server, additional security mechanisms must be provided by that particular server. This may include all of the security mechanisms of an XML firewall and any other WS security mechanism, including those recalled in section 2.6.4. (Sam Weber 2007)
- *Proxy servers*: For details on proxies and proxy servers see also section 2.6.4. Proxy Servers can be used in a SOA for different reasons according to Hutinski (Hutinski 2007b). One of them is the fact that if WSDL files are advertised with the actual locations of the service providers, an attacker could immediately find those locations. Yet, if a proxy server is used as the location in a WSDL file, the attacker will first have to bypass the proxy server in order to get to the service provider.
  - Proxy servers often have different services installed, such as XML accelerators and XML firewalls. (Heasley 2005)

---

[32] Again, the system can be distributed among different computers, or reside on a single computer. It can be equipped with different user accounts and security domains.

[33] See http://www-01.ibm.com/common/ssi/cgi-bin/ssialias?infotype=AN&subtype=CA&htmlfid=877/ENUSZG06-0353&appname=USN with examples, visited on 13-08-08

[34] See http://www-01.ibm.com/common/ssi/cgi-bin/ssialias?infotype=AN&subtype=CA&htmlfid=877/ENUSZG06-0353&appname=USN with examples, visited on 13-08-08.

- *Other XML devices*: Many different XML devices will have some security measurements installed. Some of them are hybrid hardware devices that contain a mixture of the mentioned hardware and software. It is not in the scope of this thesis to name them all.

As Hutinski (Hutinski 2007b) stated before, it is important to have security in all hardware that handles the SOAP and XML messages. This way, security measures should be found inside services, in the hardware the services are running on and in the additional hardware to upgrade the security of these services.

- **In the Service Bus or Service Broker**: If the logic of the SOA IT is concentrated around and in the Service Bus or Service Broker, then the following should security aspects should be supported by it: (see also section 2.6.4)

  "Authentication
  Authorization
  Non-repudiation
  Confidentiality
  Security standards (for example, Kerberos and WS-Security)"

  (Robinson 2004)

- **The use of security services:** IT security measures are often implemented as security services. There are different approaches for security services: one can use security services that are spread through the SOA, which will have the functionality for authentication, logon, authorisation, message filtering, et cetera. One can also create a security abstraction layer that consists of different granular security services connected to an enterprise service bus. Services such as a network firewall, application filtering, identity and access management, policy management and data security can be found in this layer. (Sluiter 2006)
- **Security build into services:** There are different development tools and platforms that can be used to create SOA services, which will have their own security measures (Mamoon Yunus 2005; W. T. Tsai 2006). These are beyond the scope of the thesis.
- **In models and documentations:** There are multiple ways of describing a service. It is actually a common trade in SOA to describe services by (multiple) models. These descriptions are often to be found as documents, or included in the service (interface) and in repositories. The following items are important for this documentation:
  - *Information model:* The information model of a service is a characterization of the information that may be exchanged with that service. In the information model, the format of the information, the structural relationships within the exchanged information and the definition of terms used are described. This helps effective use of the service by understanding more about it. It also helps security measures to understand the types of data that are expected from this service. It is necessary to have an information model for external audits as well.
  - *Consistent use of description protocols:* It is very important to be consistent in the documentation, making sure everything is formulated non-ambiguous. This helps the auditing (services or) processes. The use of open protocols that describe certain semantics ease the effort of understanding the descriptions (as a third party).
  - *Behavioural model:* Another crucial item to document is the knowledge of the actions invoked against a service and the process or temporal aspects of interacting with that service. This helps using secured objects (so one will know what needs to be done in order to access its content).
  - *The action model:* The action model is also important: it shows the characterization of the actions that may be invoked against a service. This also includes the consequents, the constraints and the policies within the use of those actions.
  - *The process model:* this characterises the temporal relationships and properties of actions and events, that are associated with interacting with the service. The further details that are necessary depend on the type of service, however that is beyond the scope of this thesis.

- *Service reachabiltiy:* It is necessary to document how a service is to be reached, what kinds of protocols are used and how it can cooperate with other services (safely).
- *Service functionality:* there should be a documentation that describes what a service does. This could include a set of constraints and policies of how to use the service.
- *The service interface:* The service interface should have a description of necessary security measures, the way of sharing information and the use of protocols, which are necessary to allow usage of the service. The interface should be well defined.
- *Policies and constraints:* there are three aspects. The policy assertion ("All messages are encrypted"), the policy owner (the one that takes the assertion and accepts it as his policy) and policy enforcement (the technology to enforce it). Policy assertions should be written in a way that is understandable to, and processed by, the parties at whom the policy is directed. A policy can cover anything, from quality of service agreements, to interface and choreography agreements as well as commercial agreements. A policy can be active at runtime, creation time or both.
- *Service contracts:* a service contract is like a policy, only more of an agreement between parties. It is a measurable assertion that governs the requirements and expectations of these parties.
  (C. Matthew MacKenzie 2006)

However, according to (Robinson 2004), it is important to assess what style of security is acceptable. They use the following questions to illustrate that:

> *"1. Is security in the communications infrastructure acceptable, for example, in the use of Secure Socket Layer mutual authentication between EAI middleware servers, or in the use of the HTTPS protocol?*
> *2. Is individual, point-to-point security acceptable between participating systems, or is an end-to-end model required? For example, is there a need to propagate client identity through intermediate systems such as brokers to the end-providers of service implementations?*
> *3. Is security in the application layer acceptable, for example, can the client code perform basic HTTP authentication with a user-id and password, or can it pass such information to the service as application data?*
> *4. Is compliance to a security standard security, such as Kerberos or WS-Security, required?"*
>
> (Robinson 2004)

Concluding:
Looking back at this section, we conclude the following for SOA and security:

- **Vision and implementation dependant:** The security measures that one may take are depending on his definition and vision of a SOA. It can be either a pure technology or business-based approach or an approach that exists as both of them.
- **SOA business security:** Multiple business security measures have to be taken into account. They are found in different fields: inside the vision, architecture (use of SOSA) and strategy (e.g. funding models), in systems, services and processes (safe reuse, service delivery life cycle, governance), in the use of training, best practices and pro-active thinking around topics such as collaboration.
- **SOA technological security:** There are multiple technological security measures for a SOA. In this thesis we have focussed on securing XML messages, which can be done by WS*- or various described protocols and standards.
- **Security for SOA is conveyed in** multiple assets, such as hardware devices, in the service bus or service broker, in security services, inside services themselves, in models and in documentation.
- **SOA security for collaboration:** There are multiple security issues for a SOA. One of the most important issues in the scope of this thesis is the lack of a well-functioning standard or security framework for SOA-based collaboration.

2.2.4. SOA concepts

Now that there is a given definition and a view on security, there is still plenty room for different implementations and ideas. To clarify what is to be expected in an organisation based on SOA, a few concepts are explained in this section. An extensive elaboration on these subjects and/or covering all of the implementations and definitions is however beyond the scope of this thesis.

- **SOI**: Service Oriented Infrastructure: A SOI provides a set of virtual hardware services that represent components, such as processors, memory, IO, devices, sensors, et cetera. (Tsai 2005) All the hardware details are hidden in the software layer. It is optimised for handling high volumes of XML traffic. SOI provides a way to manage computing resources in lockstep with application requirements both at deployment, as the workload or the requirements change. More automation would become possible by enabling devices to self-diagnose and self-repair with minimal involvement from other layers. A SOI-layer could manage the following tasks:
  - *Orchestration:* Hardware is managed to deliver a dynamic provisioning based on real time workloads and activities.
  - *Asset discovery and management:* To ensure an always up-to-date and accurate inventory of connected devices.
  - *Provisioning:* Coordinating the configuration between server, network and storage in a synchronous manner. Making sure the software is loaded in the right places, taking the right platforms in and/or out of service. Optionally providing remote booting, et cetera.
  - *Virtualisation:* Many applications can share a machine for higher utilisation rates and many machines may serve only one application for higher performance.
  - *Load balancing:* Dynamically assign resources to services, to ensure adherence in specified service levels.
  - *Capacity planning:* Measuring and tracking the consumption of virtual resources enabling to plan when to reserve them for certain workloads or when new equipment needs to be brought online.
  - *Monitoring and problem diagnosis:* Verifying that everything is operational, detecting errors and attacks in order to handle all these problems correctly.
  - *Security enforcement:* Enforcing automatic device and software authentication, identity tracing, et cetera.
  - *Logical isolation and privacy enforcement:* Ensuring fault isolation and prevention of data leaking.
  - *IT operations processes:* Setting up generic micro-IT operations as building blocks to standardise IT processes and enabling the interoperability across heterogeneous system management products.

  (Mark Chang 2006)

  SOI also stands for Service Oriented Intelligence (A.Araghi 2006) and Service Oriented Integration (Vanhanen 2003), however both are outside the focus of this thesis. On the other hand, Service Oriented Infrastructure is sometimes referred to as Infrastructure services. (Rob High 2005)
- **SOE**: Service Oriented Enterprise: A SOE, from an IT-perspective, is a model for designing software and IT infrastructure. From a business perspective, it covers the componentisation of business functions into services, whose re-composition-using business processes will result in various business functions. SOE uses a network of services in order to achieve certain objectives. (Ying Huang 2004) A SOA can consist of a SOI and a SOA, and sometimes other different elements, like the other layers named in this subparagraph.
- **SOEA**: Service Oriented Enterprise Architecture, see also SOA and SOE. (Knippel 2005)
- **SOAM**: SOA Management: A software infrastructure ensuring the production operation of SOA-based services, delivers on quality of service (QoS) expectations for technical performance, availability and, optionally, on QoS, policy management and security, business

operations, and general policy compliance. It also gives better insight in the functioning of the IT-infrastructure (James McGovern 2003). It comes in two forms:
  - *Standalone SOA Management*: SOAM-software components that can be used as a component without buying other SOA-based products from that same vendor.
  - *Embedded SOA Management*: SOAM-software components embedded in the complete software architecture inside one's SOA.

  SOA Management can be found in multiple layers of the architecture: varying from business services (service level and QoS planning), application management (security and service monitoring) to resource management (orchestration, provisioning, infrastructure health monitoring and event automation).
  (Rob High 2005; Heffner 2007)
- **Business Process management (BPM):** In order to still facilitate the functional business processes inside a company, only by the use of services, one needs to map services to the processes. This is what BPM does. (Amelia Maurizio 2008)
- **SOAG**: SOA Governance: The purpose of SOA Governance is to ensure that the overall goal of SOA —building business agility and efficiency— is effected through well-defined and consistently applied processes and policies for service planning, design, development, integration, change, deployment, and operation. The most critical objectives of SOA Governance are to achieve:
  - Coherent long-term management of your business services portfolio.
  - Appropriate, strategic design of business service interfaces.
  - A smooth evolution of your SOA platform.
  - Consistency of service implementation.
  - Control of service execution.
  - Control over decision-making processes and services.
  - Control of the service delivery life cycle.

  (Rob High 2005; Randy Heffner 2006)

  There are also some questions that arise inside SOAG, which are resolved in an implementation-specific way:
  - *Strategy and Goals:* what is governed and why?
  - *Funding, Ownership and Approvals:* who owns what? What is funded and by whom?
  - *Organisation:* what structures, processes and governance mechanisms are in place?
  - *Processes:* what are the roles, responsibilities and procedures for managing SOA activities?
  - *Policies:* what are the enforcement issues, including standards, security, release and reuse?
  - *Metrics:* what are the business outcomes, how are they measured and by what metrics?
  - *Behaviour:* what is the behavioural model, incentives, penalties and rewards for appropriate "SOA Behaviour"?

  (Liam O'Brien 2008)
- **SOA Collaboration**: SOA collaboration can be seen from multiple points of view. From a software perspective, collaboration is essential because most complex services and applications are composed of different services. From a business collaboration point of view, to integrate Business-to-Business traffic, SOA collaboration is used as well. There are different protocols and frameworks to enable collaboration of services. Notice how both XML and web services have a central role in these protocols and frameworks. (Bingnan Xiao 2006b)

2.2.5. SOA and the Collaboration Oriented Architecture: Why is it important?

A good question would be: Why would we want to use SOA? And why would a SOA need the COA framework? These two questions are answered in this section:
First we look at the reasons for using SOA. The following are found in literature:

- **Business and IT flexibility are achieved**: a more flexible enterprise architecture that is designed for continuous business change. The configuration of loosely coupled services is simple, fast and low-cost. (Stan 2008b)
- **The services are widely available for all categories of users:** the access to the services is facilitated for all users. (Stan 2008b)
- **The businesses and their processes become digital:** the new business requirements are reflected in the IT. The technology offers support and mirrors the changing business requirements. (Stan 2008b)
- **Cost Savings:** Organisations implementing SOA have the potential to achieve significant cost reductions by *reusing shareable business services*, rather than recreating functionality to address the needs of each application initiative. SOA simplifies and accelerates application development, which enables organisations to do more with less. (Stan 2008b)
- **Aligning IT to Business Processes:** SOA transforms IT systems into self-contained services that accurately reflect business processes and operational requirements. With SOA, IT mirrors business operations, which improves the utility it delivers to the business. (Stan 2008b)
- **Higher revenues:** while one can still use its legacy systems, one may also easily open new markets by exposing system features and simple redesign of the services, without having to exchange internal system features. Furthermore, customers will be satisfied because of the fast response, self-service and flexibility of the requested services. (Ohrstrom 2007)
- **SOA allows more dynamic collaboration:** One of the most important reasons is that SOA will allow more dynamic and flexible collaborations (Ismail Khriss 2007). This also includes a better and more flexible way of outsourcing (Herwig 2008). It is proven (Sanjeev Kumar 2007) that SOA does improve the performance of the complete supply chain if all parties use a proper implementation[36].

Of course, there are multiple reasons for using SOA, yet the aforementioned seems convincing enough for many enterprises to use SOA. The next question still stands though: why is the concept of COA necessary for SOA?
As one can see, SOA in itself will deliver better IT-flexibility and it allows more dynamic collaborations. It is however those dynamic collaborations that are problematic without using the elements of the COA framework. One has already tried using security frameworks (See also section 0) to enable a safe collaboration between parties, yet these efforts are without global acceptance or a complete success. A variety of problems around collaborations based on SOA remain, that are not easily resolved. Examples are the dissolving perimeters and the need for trust. That is why the COA framework is necessary to get full benefits from a SOA. See also chapter 3.

2.2.6. Concluding: What is SOA and where do we focus on?

Looking back on this paragraph, we conclude the following:

[36] There are a few other requirements for supply chain performance as well, but that is outside the scope of this thesis.

- **SOA:** SOA stands for Service Oriented Architecture, which is an architecture based on services. These services provide a way of shaping business and technology. Services themselves have certain properties and are provided by service providers, which deliver them through service brokers to the service requesters, which in turn become service consumers.
- **SOA is not a solution by itself**: SOA alone is not a solution; it is a paradigm that enables the user to get more value from the local and foreign, as in: under control of others, capabilities.
- **SOA has many security measures:** There are many security measures on both the business side as well as the IT-security side. The business measures focus on the correct vision, governance, processes, lifecycles, training, best practices, collaboration, roadmaps and strategy. Both IT and business security measures are very implementation specific. If a SOA is 'web service and XML'-based, there are many security measures based on W3C and OASIS recommendations and standards.
- **SOA still has a lot of security issues:** SOA is far from perfect: there are many security issues that organisations have to cope with by themselves. One of the most important issues is how to collaborate. Even though there are many frameworks and measures devised, none of them seem to be the standard.
- **SOA security can be found in...**: SOA security measures can be found in specific hardware, the service broker, security services, inside services and in models and documentation that reside in documents, service interfaces and/or repositories.
- **Other SOA concepts:** In this paragraph we have seen:
  - *SOI (Service Oriented Infrastructure):* A SOI provides a set of virtual hardware services that are optimised for high volumes of XML traffic. It provides a way to manage computing resources and manages a variety of tasks for high quality IT-management and hardware management processes.
  - *SOE (Service Oriented Enterprise):* a model for designing software and IT infrastructure. It also covers the business componentisation of functions into services.
  - *SOAM (Service Oriented Architecture Management):* A software infrastructure ensuring the production operation of SOA-based services delivers on the right conditions and the right qualities.
  - *Business Process Management:* A way of modelling and mapping out the services to the functional business processes.
  - *SOAG (SOA Governance):* The purpose of SOA governance is to ensure that the overall goal of SOA —building business agility and efficiency— is affected.
  - *SOA Collaboration:* From a software perspective, it is about sharing services in order to create applications or complex services, from a business perspective it is about collaboration between parties.
- **SOA has many benefits:** For example better business and IT-flexibility, service availability, cost savings, better business-IT alignment, higher revenues and more flexible and dynamic means of collaboration.
- **SOA needs the COA framework for more secure collaboration:** The current flexible and dynamic collaborations create many security problems. Even though security frameworks and measures have already been built, there still is no appropriate standard for resolving all these issues, such as the dissolving perimeters and the need for trust. The COA framework shall be the standard for resolving all these problems.

These conclusions are imperative for two reasons: First, knowing some of the issues of and the security in a SOA, we can now see the value that the COA framework will add. Second, we can see what to expect when we want to use the COA framework for another entity to make it COA framework-compliant. Third, we found an architectural vision of loosely coupled services, that we see again in the architectures in chapter 4.
This automatically means that we will use the information we have researched in both chapter 3 through chapter 5. There is no real focus for SOA; we have to use all of the information to answer our research questions, though. This should not surprise, since SOA is part of the basics.

2.3.Introduction to Software as a Service

2.3.1.Introduction

An important way of collaborating by using the principles of SOA can be found in Software as a Service (SaaS). It is one of those special services where security is needed and valued more, and de-perimeterisation comes more into play. In fact, as we will see in chapter 4, it can be a very important item for a COA or for entailment to the COA framework. This is why we find SaaS as being a part of the basics in Figure 3.

In this paragraph we look at SaaS to see what it is (section 2.3.2) and what kind of security issues lie within it(section 2.3.3). We summarise the information given in this paragraph in section 2.3.4 and set our focus for SaaS as well.

2.3.2.Definitions: SaaS as we see it

Metsaars in (Metsaars 2008b) used a good definition and added an analysis to it:

> *"Software as a Service (SaaS) is time and location independent online access to a remotely managed server application, that permits concurrent utilization of the same application installation by a large number of independent users (customers), offers attractive payment logic compared to the customer value received, and makes a continuous flow of new and innovative software possible"*
>
> ***Time and location independent online access***
> *A characteristic of SaaS is that the software is offered over the Internet (online access). It does not matter where a user/customer is or what time it is, when one wants to use the software. If one has the ability to connect the Internet with an Internet browser, then it is possible to use the software.*
>
> ***Concurrent utilization of the same application installation***
> *SaaS software is built as a configurable and scalable service that uses metadata to provide different experiences to customers. Software offered like this is defined as 'multitenant'. A multitenant system is comparable to a block where a number of different flats share land, stairs, roof and so on. This make it possible to offer concurrent utilization of the same application installation.*
>
> ***Attractive payment logic compared to the customer value received***
> *With SaaS, software is offered on a subscription or lease basis rather than as a packaged product to purchase and bring in-house. The customer pays a fixed fee for a determined period based on which features of the software one uses. For instance, a one-man business needs the financial part of an ERP system to do its bookkeeping. Yet, it is far too expensive to buy the whole ERP system. Instead, it is offered to only use and pay for the financial part of the ERP software.*
>
> ***Makes a continuous flow of new and innovative software possible***
> *If software is offered as a service, it can be combined with other software (services) and offered as new and innovative software..."*
>
> (Metsaars 2008b)

Further research shows, that SaaS is not exactly new. It has been known as ASP for quite a while (Dirk Hanenberg 2008). The basic concepts derive from the application service provider, as described by IDC in one of their whitepapers in 1999. The basic principles were back then: An application service provider works application-centric, provides only the application access (no ownership), that is centrally managed, based on a one-to-many service that is delivered on contract. In this original ASP concept the customer could, besides remote usage of application software, also rent the required data centre infrastructure and application support. The ASP concept was offered as a new and competitive form of remotely hosted and outsourced application service for the customers. One could say that SaaS and ASP are about application outsourcing. (Markku Sääksjärvi 2005)

Because of the application outsourcing principle, some aspects, like the users, differ from traditional applications. Metsaars described this as:

> *"The concept of "users" is, in a SaaS application, a bit more complicated than in traditional applications. With SaaS applications, there are tenants, which are the organisations that use the application to access their own data store. This data store is logically isolated from data stores belonging to any other tenants. The employees of the tenant are the end users, who are granted access to the application by the tenant, allowing them to access some portion of the tenant's data. The data that is available for end users must be filtered because different end users have different data needs and not every end user may see all the business critical information."*
>
> (Metsaars 2008b)

Later on, the Software & Information Industry Association presented arguments for considering the next generation ASP model as a service concept. They introduced the idea of Software as a Service. SaaS fulfilled the IDC definition of ASP and went even further; besides standard application, little customisation to that applications was done. (Markku Sääksjärvi 2005)

Some argue that SaaS does differ from ASP, because SaaS allows a larger supplier network and will work with a much finer granularity of software modules that are more customisable. (Nicolas Gold 2004)

There are different business models to use ASP (see (Dirk Hanenberg 2008)) and there are different perspectives and business drivers for SaaS (see (Metsaars 2008b) for more details), yet this is out of the scope of this thesis.
In this thesis we look at SaaS as a software service that is offered at the internet by another party, based on the ideas of SOA.

2.3.3. SaaS and security

Security issues and prevention:
Since the COA framework is an information security framework, it would be a good idea to see what kind of security issues there are with SaaS, and what kind of measures are already used. This allows us to see what value the COA framework could bring to SaaS and how it can be used for modern collaborations:

Metsaars described multiple security threats, risks, security objectives, services, patterns, mechanisms, patterns, components, models and guidelines (Metsaars 2008b) for SaaS. Here is a summary of that work, as far as the objective of the thesis is concerned (see (Metsaars 2008b) for all the details):

- **Security risks for SaaS:** the following have been named in relation to SaaS:
  - *Identity theft:* the attacker steals an identity form an end-user of the software service, which makes it possible for the attacker to access critical business data.

  - *Exposing sensitive and private data*: The attacker has the ability to touch critical business data.
  - *Application unavailability*: The application becomes unavailable for the customers.
- **Security threats (and protection mechanisms):** the following threats and protection mechanisms have been named in relation to SaaS:
  - *The threats:* most of the common threats comply here as well, like eavesdropping, DoS attacks, unauthorised access, Man-in-the-middle attacks, Trojan Horse programs, viruses, worms and phishing attacks (see (Metsaars 2008b) for a complete outline and more details).
  - *The protection:* The following security mechanisms were named: Encryption, techniques against DoS, based on hardware or software, authentication, digital certificates, encryption and updated antivirus software. (See also section 2.6.4 and (Metsaars 2008b) for more details)
- **Security objectives:** The following security objectives have been named in relation to SaaS security:
  - *Identification:* Ensure end-users and data are identified and that their identities are properly verified;
  - *Authorization:* Ensure end-users and client applications can only access data and services for which they have been properly authorised;
  - *Confidentiality:* Ensure confidential communications and data are kept private;
  - *Data protection:* Ensure communications and data are not intentionally corrupted;
  - *Non-repudiation:* Ensure parties in interactions with the application or component cannot later repudiate those interactions;
  - *Application availability:* Ensure the application satisfies the previous six objectives to prevent attacks and downtime.
- **Security patterns:** The following security patterns have been named as being necessary for SaaS:
  - *Single point of access*: when there are multiple doors to enter an application, supervision of access to that application becomes hard. This is why, with the help of a checkpoint (and a session), the application can be accessed via one way.
  - *Checkpoint*: when one wants to use a single point of access, security should be enforced. In fact, only those who are authorised should enter that point. A checkpoint will be a security enforcing object that acts as a gateway to the application's single point of access and it will restrain those not allowed, for any reason, from that application.
  - *Roles:* Since it is hard to define for every single user what their rights and obligations are for a certain application, one can define objects that represent roles, which in turn define permissions and access rights that groups of users have.
  - *Session*: Secure applications often need access to shared values, but these values are not unique throughout the system (such as login credentials from users and their respective roles). A session object provides a common interface for all components and/or applications to access the sensitive information.
  - *Filtered view:* Users should not be allowed to perform illegal operations by changing data they are not supposed to see (see also section 2.3.2). In order to make sure a user cannot access any data besides the data he has been privileged to by the roles, a filtered view ought to be created, restricting the graphical user interface to the information to which access is granted.
  - *Secure access layer:* if the security is not properly integrated with that of the external systems it uses, applications become insecure. That is why, if necessary, one should have a set of lower security mechanisms that act as a secure access layer for communicating in and out the program.
- **Security services:** The security services, as named in section 2.6.4 also apply to SaaS.
- **Security measures:** The following measures can be found in SaaS:
  - *Firewall:* see section 2.6.4 for more details.

- *Authentication server:* An authentication server makes sure that the users who access the application are who they claim to be.
- *Application server:* See also section 0.
- *Backup server:* A backup server makes sure that tenant data is not totally lost in case of data loss. This data needs proper security services to be secured.
- *Database server:* The database server holds all the information of the tenant, which needs proper security. Various security services are necessary to secure the data on this server.
- *Antivirus software:* Antivirus software is necessary to protect against malware (viruses, worms, Trojan Horses and other related security threats).
- *Security policies:* See also sections 2.6.5, 2.6.9.

(Metsaars 2008b)

As visible, some of the details of the summary are changed into references into other domains, like SOA and the Jericho related security measures. This is because those security mechanisms exactly look alike.

SaaS and COA:
One of the objectives that Metsaars defined in (Metsaars 2008b) was the use of SaaS collaboration, which consists of the creation of new software within existing software by reusing and combining them. Collaboration in SaaS is important; it is in fact one of the basics of SaaS. Not just for creating new software; even when one wants to use SaaS without recomposing the software, one will have to cooperate with the software vendor.
Yet, in (Dirk Hanenberg 2008) we saw that SaaS encountered a trust problem which is being partly resolved by the growing customer base. Reasons behind the trust issues are variable. The automated negotiation between parties and the distributed nature of SaaS are two of them. (Nicolas Gold 2004)

In short:
With collaboration, the security issues and the lack of trust taken in mind, one may argue that the COA framework looks like a promising candidate to improve the usage of SaaS. We will see how in chapter 3.

2.3.4. Concluding: What is SaaS and where do we focus on?

Looking back on this paragraph, we come to the conclusion that:

- **SaaS**: SaaS stands for Software as a Service. It is time and location independent online access to a remotely managed server application, which permits concurrent utilisation of the same application-installation by a large number of independent users. It offers attractive payment logic and makes a continuous flow of new and innovative software possible.
- **Former ASP:** SaaS has been known as ASP for quite a while (since 1999). It was then centrally managed, access-only, one-to-many and delivered on a contract.
- **Difference from ASP:** Some argue that SaaS and ASP are the same, while others claim that SaaS has a finer granularity, is more customizable and has a larger supplier network.
- **SaaS and security risks and threats:** SaaS has the following security risks and threats:
  - *Risks:* identity theft, exposing sensitive and private data and application unavailability.
  - *Threats:* eavesdropping, DoS attacks, unauthorised access, Man-in-the-middle attacks, Trojan Horse programs, viruses, worms and phishing attacks.
- **SaaS and security measures:** SaaS uses different security objectives, patterns, services and measures. Most of them can be found in other fields as well.
- **SaaS and trust:** a considerably relevant issue for SaaS is trust. This is because of the automated negotiation between parties and the distributed nature of SaaS.

Knowing what SaaS is and having some insight in SaaS and its security and (lack of) issues around trust, we can set the focus for SaaS and this thesis.
We will use SaaS in chapter 3 as an example of a collaborating party over the internet, which will collaborate with a COA based on the elements of the COA framework for a secure and trustworthy collaboration between the parties.

2.4.Control Objectives for Information and related Technology

2.4.1.Introduction

In order to control the COA framework within his Collaboration Oriented Architecture, one needs proper governance. We have seen some matters of governance embedded within SOA. This is not enough, though. The Control Objectives for Information and related Technology (COBIT)-framework is a very important addition. COBIT comes from the IT Governance Institute (ITGI™) that was established in 1998 to advance international thinking about (and standards in directing and controlling) an enterprise's information technology. The COBIT Mission states:

> *"To research, develop, publicise and promote an authoritative, up-to-date, internationally accepted IT Governance control framework for adoption by enterprises and day-to-day use by business managers, IT professionals and assurance professionals"*
>
> (Institute 2007b)

This paragraph shows what IT Governance and the COBIT framework are all about. We will look at:

- IT Governance (section 2.4.2), the focus of IT Governance (section 2.4.3) and its advantages (section 2.4.4)
- COBIT (section 2.4.5), the use of COBIT for this thesis (2.4.6) and the focus for this thesis considering COBIT (section 2.4.7)

Section 2.4.7 will be the concluding and summarising section of this paragraph. It is imperative to understand that COBIT and/or Governance is a important basic. See also Figure 3.

Intermezzo 2: IT Governance and process control explained by the PDCA.

IT Governance is an important aspect of Information Security (IS). In order to govern the processes, one could use the PDCA (Plan Do Check Act), (Standardization 2005) also called the Deming Circle. The PDCA circle for IT Governance consists of:

- **Plan:** Planning by creating the strategy and tactics that are used to ensure IT will contribute to business objectives in the right way.
- **Do:** One has to develop his processes, solutions and controls[1], and implement them.
- **Check:** One has to evaluate whether they are functional.
- **Act:** One has to alter its processes and controls, based on the outcomes of the evaluation (check).

This circle shows what IT Governance in a nutshell is all about.

[1]: See section 2.4.5 for more details on auditing and controls.

2.4.2.IT Governance:

As the mission states, COBIT is about creating an IT Governance control framework. So, what is IT Governance? The ITGI defines this in its executive overview of COBIT 4.1 as:

> *"IT governance is the responsibility of executives and the board of directors, and consists of the leadership, organisational structures and processes that ensure that the enterprise's IT sustains and extends the organisation's strategies and objectives."*
>
> (Institute 2007b)

In principle, IT Governance is in close relation with Information security:

> *"Information security is the protection of information from a wide range of threats in order to ensure business continuity, minimize business risk, and maximize return on investments and business opportunities…" "…Information security is achieved by implementing a suitable set of controls, including policies, processes, procedures, organisational structures and software and hardware functions. These controls need to be established, implemented, monitored, reviewed and improved, where necessary, to ensure that the specific security and business objectives of the organisation are met. This should be done in conjunction with other business management processes."*
>
> (joint technical comittee ISO/IEC JTC 1 2005)

Furthermore, IT Governance integrates and institutionalizes good practices to ensure that the enterprise's IT supports the business objectives. IT Governance enables the enterprise to take full advantage of its information, thereby maximizing benefits, capitalising on opportunities and gaining competitive advantage. (Institute 2007b)
IT Governance comes down to controlling the IT-processes, their performance, maturity, risks, resources and value (See also section 2.4.4). This is done by a proper usage of different frameworks and measures.
There are various models that measure (and could advise about) the performance of different aspects inside an enterprise:

- **Balanced Scorecard (BSC):** It is a model designed by Kaplan and Norton. It is a coherent set of performance measures organised into four categories. It includes financial, customer, internal business processes and learning and growing perspectives. (Institute 2007a)
- **Capability Maturity Model (CMM):** The CMM is designed at the Software Engineering Institute (SEI) and allows one to measure the maturity of a process or other objects. It contains five levels of maturity. There are many derived standards from the CMM such as the CMMI and SECMM. It is also used in an alternative form in COBIT.(Institute 2007a)
- **Six Sigma:** A methodology provides a business with the tools to improve the capability of their business processes. Its goal is to increase profits by eliminating variability, defects and waste.[37]

Another important set of measures and frameworks are developed with a focus on ICT service management. These are created in order to establish and maintain efficient and effective ICT (-support and -management) services and processes. The following measures and frameworks have been often used:

- **ISO 20000:** Former BS15000, one of the first IT service management standards. It comprises a specification (ISO/IEC 20000-1:2005) for a service management system and a code of practice (ISO/IEC 20000-2:2005). They are aligned with the process approach within ITIL.[38]
- **ITIL:** ITIL is best practice in IT Service Management, developed by OGC and supported by publications, qualifications and an international user group. ITIL is intended to assist organisations in developing a framework for IT Service Management. It consists of a series of

[37] See http://www.isixsigma.com for more details, visited on 14-08-08.

[38] See http://www.isoiec20000certification.com/about/whatis.asp and http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=41332 for more details, visited on 14-08-08.

libraries giving guidance on the provision of quality IT services, as well as on the accommodation and environmental facilities needed to support IT.[39]

Furthermore, there are sets of frameworks, which are aimed on IT (and business) security and governance:

- **COBIT:** Control Objectives for Information and related Technology. See section 2.4.5 for more details.
- **Internal Control – Integrated Framework:** A framework from COSO, which is a control framework for enterprise governance and risk management. COBIT has been designed to fit into this framework.[40] (Institute 2007b)
- **ISO 27002:** ISO/IEC 27002:22005 Information technology - Security techniques - Code of practice for information security management. A standard of good practice for information security. It started as BS 7799(formerly known as DTI Code of Practice for Information Security Management) in 1995 and has evolved over the years into this standard. It is focused on information security (instead of only IT security). It is an advisory document that lays out a well-structured set of suggested controls to address information security risks. Many access control policies are based on ISO 27002. It also gives 39 control objectives.[41]
- **ISO 27001**: ISO/IEC 27001:2005 formerly known as BS7799. It specifies the requirements for establishing, implementing, operating, monitoring, reviewing, maintaining and improving a documented Information Security Management System, within the context of the organisation's overall business risks. It specifies requirements for the implementation of security controls customised to the needs of individual organisations or parts thereof. It also includes the PDCA (Plan Do Control Act) –circle.[42]
- **SABSA:** Sherwood Applied Business Security Architecture. It is a proven framework and methodology for Enterprise Security Architecture and Service Management. It considers many things among which are: Risk Management, Information Assurance, Governance and Continuity Management. It is developed to be the umbrella over all the other frameworks, such as COBIT.[43]

Another interesting measure is Prince2, which is used for project management: Projects in Controlled Environments 2, its former version PRINCE, was introduced as a UK government standard for IT project management. PRINCE2 is a standard on project management and provides a structured framework, which can be tailored and scaled in order to run a project.[44]

The exact details of the frameworks are out of the scope of this thesis.
Besides the ITGI, there are a few other organisations as well that focus on either information security and/or (IT) governance:

- **COSO:** Committee Of Sponsoring Organisations of the Treadway Commission[45]: They have developed a few guides and frameworks focussed on internal control, enterprise risk management and financial reporting. Their most important framework is the Internal Control

[39] See http://www.itil-officialsite.com for more details, visited on 14-08-08.

[40] See http://www.coso.org/ for more details, visited on 07-04-08

[41] See http://www.iso27001security.com/html/27002.html and http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=50297 for more details, visited on 14-08-08.

[42] See http://www.27001-online.com/ and http://www.iso.org/iso/catalogue_detail?csnumber=42103 for more details, visited on 14-08-08.

[43] See http://www.sabsa.org for more details, visited on 14-08-08.

[44] See http://www.prince-officialsite.com for more details, visited on 14-08-08.

[45] See for a historical overview on COSO: http://www.coso.org/aboutus.htm , visited on 14-08-08.

– Integrated framework, which is an enterprise governance and risk management framework. ITGI has designed COBIT in such a way, that it should fit exactly with this framework. [46]

- **ISACA:** The Information Systems Audit and Control Association[47]: An organisation that focuses on information governance, control, security and audit professionals. It has IS auditing and IS control standards, a certification system for personnel and a technical journal. [48]
- **ISO:** International Organisation for Standardization is the world's largest developer and publisher of international standards. It has numerous bodies, members, committees and working groups. It had over 650.000 standards published at the end of 2007, comprising many different fields and topics. What makes this organisation interesting, is the large scale of IT-standards and standards on IT Governance, service management and information security.
- **IEC:** International Electrotechnical Commission: It is an international commission for standards for electrical, electronic and related technologies. It works together with the ISO on many other standards.[49]
- **itSMF:** the IT Service Management Forum, which is a community for IT-(service) management and – strategy. It works closely with OGC and other groups. It is one of the organisations behind publications such as ITIL V3 and other standards and frameworks.[50]
- **OGC:** Office of Government Commerce is an independent office of HM Treasury, established to assist Government in delivering best value from its spending. The OGC works with central UK Government departments and other public sector organisations to ensure value for money and delivery of projects, the government estate and more. It is one of the organisations behind ITIL and other frameworks and standards.[51]
- **SABSA Limited:** this is the governing body for the SABSA Method and the intellectual property embodied in it. SABSA Limited governs the use of SABSA intellectual property by qualified individual architects and consultants, and by licensed education providers internationally.[52]

The exact details of the organisations are out of the scope of this thesis.
IT Governance became a more important topic, when the Sarbanes-Oxley act (SOX) was stated during the 107th congress of the United States of America. Its focus:

> *"To protect investors by improving the accuracy and reliability of corporate disclosures made pursuant to the securities laws, and for other purposes"*
>
> (Sarbanes 2002)

The idea behind SOX is that organisations have to be in control of the processes inside their organisation and thus need to audit them.

Concluding: IT Governance
Looking back at this section, one can say that:

[46] http://www.coso.org/ , visited on 07-04-08

[47] See for a historical overview on ISACA: http://www.isaca.org/Content/NavigationMenu/About_ISACA/Overview_and_History/Overview_and_History.htm visited on 14-08-08.

[48] http://www.isaca.org/ for more details, visited on 07-04-08.

[49] See http://www.iec.ch/ for more details, visited on 14-08-08.

[50] See http://www.itsmf.org/ for more details, visited on 14-08-08.

[51] See http://www.ogc.gov.uk/ for more details, visited on 14-08-08.

[52] See http://www.sabsa.org/ for more details, visited on 14-08-08.

- **IT Governance** is strongly related to information security. It consists of leadership, organisational structures and processes that ensure an enterprise's IT sustains and extends the organisation's strategies and objectives.
- **Models and measures related to IT Governance**: there are models and measures that are related to IT Governance, such as the BSC, CMM, ISO 20000, ITIL, COBIT, ISO27000 series. These measures come from different organisations, such as the ITGI, ISACA, ISO.

2.4.3. Advantages of IT Governance:

There are a lot of advantages of - and drivers for IT Governance:

- **Strategic Alignment of I(T) with business:** aligning information security with business strategy to support organisational objectives. (Rogelio Aguilar Alamilla 2008)
- **Risk management:** Executing appropriate measures to mitigate risks and reduce potential impacts on information resources to an acceptable level. (Rogelio Aguilar Alamilla 2008)
- **Value delivery:** Optimising security investments in support of business objectives. (Rogelio Aguilar Alamilla 2008)
- **Resource management:** Using information security knowledge and infrastructure efficiently and effectively to ensure availability and correct processing of information, as well as to ensure the proper utilisation of the infrastructure. (Rogelio Aguilar Alamilla 2008)
- **Performance Management:** Monitoring and reporting on information security processes to ensure that objectives are achieved. (Rogelio Aguilar Alamilla 2008)
- **Integration of assurance:** Integrating all relevant assurance factors to ensure that processes operate as intended from end to end. (Rogelio Aguilar Alamilla 2008)
- **Take full advantage of information:** ITGI shows that IT Governance will enable enterprises to take full advantage of their information, thereby maximising benefits, capitalising on opportunities and gaining competitive advantage (Institute 2007b).
- **Maximise transparency on IT:** One of the most important drivers behind IT Governance is the lack of transparency of It's costs, values and risks (Institute 2007b).

Much more can be said about the advantages, however it is out of the scope of this thesis to specify all of the advantages of IT Governance.

2.4.4. IT Governance focus:

The ITGI summarises the governance focus areas as follows:

> *"**Strategic alignment** focuses on ensuring the linkage of business and IT plans; defining, maintaining and validating the IT value proposition; and aligning IT operations with enterprise operations.*
>
> ***Value delivery** is about executing the value proposition throughout the delivery cycle, ensuring that IT delivers the promised benefits against the strategy, concentrating on optimising costs and proving the intrinsic value of IT.*
>
> ***Resource management** is about the optimal investment in, and the proper management of, critical IT resources: applications, information, infrastructure and people. Key issues relate to the optimisation of knowledge and infrastructure.*

Figure 5: IT Governance focuses. (Institute 2007a)

Risk management *requires risk awareness by senior corporate officers, a clear understanding of the enterprise's appetite for risk, understanding of compliance requirements, transparency about the significant risks to the enterprise and embedding of risk management responsibilities into the organisation.*
Performance measurement *tracks and monitors strategy implementation, project completion, resource usage, process performance and service delivery, using, for example, balanced scorecards that translate strategy into action to achieve goals measurable beyond conventional accounting."*

(Institute 2007b)

Intermezzo 3: Auditing: what is it?

"An audit is simply a review of past history…" "...We will use the following concatenated definition:.." "...Audit is a systematic process of collecting evidence to test or confirm a statement or to confirm a record of transaction…""…there are two basic types of audits: one that verifies compliance (compliance test) and one that checks the substance and integrity of a claim (substrantative test)"

(David L. Cannon 2006)

In an audit, there is always an auditor (he who performs the audit, can be internal or external), and the auditee (he who is being audited). There are several types of audits such as financial, operational, integrated, compliance, administrative and information systems audits.
Each of the audit types has its own audit standard and (often) framework. See (David L. Cannon 2006) for more details.
(David L. Cannon 2006)

2.4.5. COBIT:

Now that IT Governance is clarified a little, we can focus on what COBIT is. We will focus only on the main lines and some details of important processes. It is beyond the scope of this thesis to fully outline the necessity of a control framework, or to specify the exact details of all elements provided by the COBIT Framework. One may look that up if inclined to understand in (Institute 2007a).

General description:
The ITGI describes COBIT as:

"COBIT is a framework and supporting tool set that allow managers to bridge the gap with respect to control requirements, technical issues and business risks, and communicate that level of control to stakeholders. COBIT enables the development of clear policies and good practice for IT control throughout enterprises. COBIT is continuously kept up to date and harmonised with other standards and guidance. Hence, COBIT has become the integrator for IT good practices and the umbrella framework for IT governance that helps in understanding and managing the risks and benefits associated with IT. The process structure of COBIT and its high-level, business-oriented approach provide an end-to-end view of IT and the decisions to be made about IT."

(Institute 2007a)

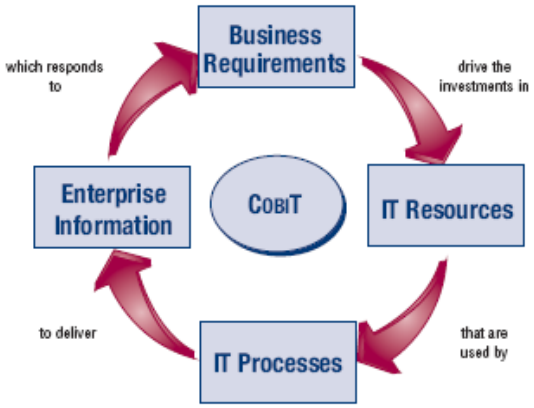


Figure 6: The main principle of COBIT (Institute 2007a)

Main Principle:
It is based on the following principle:

> *"To provide the information that the enterprise requires to achieve its objectives, the enterprise needs to invest in and manage and control IT resources using a structured set of processes to provide the services that deliver the required enterprise information."*
>
> (Institute 2007a)

This is done by:

> ***"Benchmarking** of IT process performance and capability, expressed as maturity models, derived from the Software Engineering Institute's Capability Maturity Model (CMM).*
>
> ***Goals and metrics** of the IT processes to define and measure their outcome and performance based on the principles of Robert Kaplan and David Norton's balanced business scorecard.*
>
> ***Activity goals** for getting these processes under control, based on COBIT's control objectives."*
>
> (Institute 2007a)

Focus of the thesis considering COBIT

The focus of the thesis will be on processes and the Control Objectives, where the first priority will be on the process "Manage Third-party Services", which is an element of the "Deliver and Support" domain

Components and Relationships:

The COBIT Framework consists of multiple components and relationships, as visible in Figure 9. The top of the figure is explained as follows: In order to use IT properly, one first has to organise a set of business goals, from which the business goals and a strategy are established. These goals and the strategy, along with COBIT-information criteria[53], will generate a set of business requirements or business goals for IT. These can be translated to IT goals, which in turn are used to design the Enterprise Architecture for IT. Once the business and IT goals have been defined, they need monitoring to ensure the actual delivery matches expectations. This is achieved by metrics that derive from the goals and are captured in an IT scorecard. (Institute 2007a)

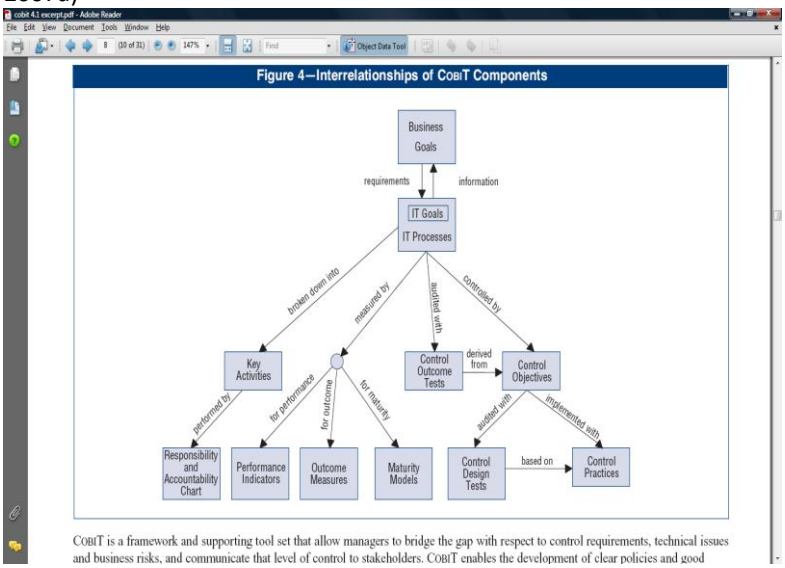


Figure 7: The COBIT Framework. (Institute 2007a)

Processes & Domains:

The focus of COBIT is on the IT processes: There are 34 IT processes divided into four domains to provide an end-to-end view of IT. The processes combined together are delivering information in line with the IT goals. They deliver information by running applications, requiring infrastructure and people. The IT goals are focused on fulfilling the business goals, while they are influenced by governance requirements. The four domains are called:

- **Plan and Organise (PO)**: covers strategy and tactics (setting up a strategy, communicating it and manage it) and focuses on how IT can contribute to the achievement of business objectives.

[53] These are out of the scope of the thesis, see Institute, I. G. (2007a). COBIT 4.1. Framework Control Objectives Management Guidelines Maturity Models. Rolling Meadows, IT Governance Institute. For more details.

- **Acquire and Implement (AI)**: is about developing or acquiring IT solutions and the implementation and integration into the business process. Changing and maintaining these solutions is also part of the domain.
- **Deliver and Support (DS)**: concerns with the actual delivery of required services. It includes service delivery, management of security and continuity, service support for users, and management of data and operational facilities.
- **Monitor and Evaluate (ME)**: covers performance management, monitoring of internal control, regulatory compliance and governance.

It is interesting to see that the named domains can actually be mapped to the PDCA circle, which is seen often in IT Governance, as seen in intermezzo 2:

PDCA	*COBIT*	*PDCA*	*COBIT*
Plan	Plan and Organise	Do	Acquire and Implement & Deliver and Support
Check	Partial Monitor and Evaluate	Act	Partial Monitor and Evaluate

Figure 8: Mapping between PDCA and COBIT domains.

COBIT provides examples for each process as illustrated by:

> *"-Generic inputs and outputs.*
> *-Activities and guidance on roles and responsibilities in a Responsible, Accountable, Consulted and Informed chart.*
> *-Key activity goals (the most important things to do)*
> *-Metrics"*
>
> (Institute 2007a)

Control (Objectives):
Each process needs controls. (Policies, procedures, practices and organisational structures to provide reasonable assurance that business objectives will be achieved and undesired events will be prevented or detected and corrected) (Institute 2007a).
There are general controls (embedded in IT processes and services: development, security, et cetera) and application controls (embedded in business process applications: completeness, accuracy, validity, et cetera).
The design and implementation of application controls are the responsibility of IT, covered in the AI domain. Business is also responsible for them, because they have to define functional and controlled requirements. The IT is completely responsible for general IT controls.
The size of a control ranges from a complete process to a single activity: the basic idea is that a control has to support a control objective. If a control objective would be "The prevention of alterations to the system by people unauthorised to do so.", then many controls can be used to set this up: from security procedures to actions like setting up a firewall.

COBIT provides a set of IT control objectives, which provide a complete set of high-level requirements to be considered by management for effective control of such an IT process. They are focused on either increased value or reduced risk. They contain a control thanks to fewer errors in the process and a more consistent management approach.
The control objectives are identified as a domain reference (PO, AI, DS and ME) plus a process number and a control objective number. There are also generic control requirements for processes, named as PCn, where n is a number.
Enterprise management needs to choose if they want to implement it. If they do; how they will do it, and if they do not; they have to accept the risk of not implementing those that could be

important. They can easily see whether they are important or not because there is a clear relation between IT Governance requirements, IT processes and IT controls.
Control Objectives are audited with Control Design Test to see whether the Control Practices, are implemented correctly for the Control Objectives.

Measurements: Maturity models, Performance goals and metrics, Activity Goals
As one can see in Figure 9 on page 48, there are also three ways of measuring the system:

> *"Maturity models to enable benchmarking and identification of necessary capability improvements*
> *Performance goals and metrics for the IT processes, demonstrating how processes meet business and IT goals and are used for measuring internal process performance based on balanced scorecard principles*
> *Activity goals for enabling effective process performance"*
>
> (Institute 2007a)

These three ways are imperative because an enterprise needs to understand the status of its own IT systems and decide what level of management and control the it should provide, taking the status (that is, the results of the measures) and the IT objectives into that calculation.
However, these measurements are outside the scope of the thesis as well: we are more interested in the control(-s and) objectives.

Responsibility and Accountability Chart:
Once the controls are defined, they can be committed to either an individual or an organisation. They will be responsible for the results of that control. Problems, controls and activities will become feasible by the organisation. One will be capable of delegating a control or an activity and control or direct that as well. Thus, as soon as anything goes wrong, the appointed responsible party will have to take this into account and fix the problem.

Concluding: COBIT
We conclude by stating that COBIT is an IT Governance framework, which consists of multiple elements such as processes, control (objectives), maturity models, performance goals and metrics, activity goals, the responsibility and accountability chart et cetera. Its main goal is to provide support in IT Governance.

2.4.6. Use of COBIT for this thesis

As noticed in the introduction (See 1.1), COBIT will be considered in order to apply governance in a COA. Governance is elemental in such an enterprise that is a COA! If one wants to fully trust a third party, then that third party should be fully in control of its assets, processes, et cetera. That is why governance is so important.
It would be unwise to fully describe every single detail of the governance itself inside the reference architecture, because every implementation is different and thus the applied governance will be different.

The thesis will experiment only with the process "DS2 Manage third-party services", that is an element of the DS domain (see marked element on figure 4).
One might ask himself why the DS2 service is prioritised? This is because it is one of the most important processes on the field of collaboration. It helps to manage the services of third parties, say, other SOA-based organisations that deliver their materials via SaaS or another way of sharing services.
If one wants to know which other control objectives and respective processes could be used, look at intermezzo 4.

2.4.7. Concluding: What is COBIT and where do we focus on?

Looking back at this paragraph, we state the following:

- **COBIT:** stands for Control Objectives for Information and related Technologies and comes from the IT Governance Institute. It is a framework and supporting tool set that allows managers to govern over one's IT. It allows one to deal with control requirements, technical issues and business risks by policies, controls (objectives), processes and best practices. It gives an end-to-end view of IT and the decisions that have to be made about IT. Based on the principle to provide information an enterprise requires to achieve its objectives, it helps that enterprise to invest and manage IT resources. This principle is reached through benchmarking, (activity) goals and metrics.
- **IT Governance:** IT Governance is focused on ensuring the enterprise's IT sustains and extends the organisation's strategies and objectives. This is done by multiple processes, organisational structures, leadership, best practices and measures, and frameworks like COBIT.
- **COBIT is one of the (IT-) Governance measures:** Beside COBIT exists a variety of other measures and frameworks, like the Balanced Scorecard, the CMM, the Internal Control-Integrated framework, several ISO standards, ITIL, SABSA, Six Sigma and Prince.
- **IT Governance delivers many advantages:** IT Governance delivers advantages such as better strategic alignment of IT with business, risk management, better value delivery, better resource and performance management, integration of insurance factors and a maximum transparency on IT.
- **IT Governance focuses on** strategic alignment, value delivery, resource management, risk management and performance measurement.
- **The COBIT Framework consists of:** There is a set of business goals, translated to IT goals that will be achieved by the usage of IT processes. The processes are spread among four domains: Plan and Organise, Acquire and Implement, Deliver and Support and Monitor and Evaluate. Each process is accompanied by Control (Objective)-s, maturity models, performance goals and metrics, activity goals and a responsibility and accountability chart.

Knowing what COBIT and IT Governance are about, we can define a focus. In this thesis we focus on the process DS2 "Managing third party services". We will look at the process and the including control (objective)-s. See chapter 3 for more details.
However, one should understand that this is only one of the many governance frameworks that is necessary in a COA. Those however, are out of the scope of this thesis in order to make it feasible for now.

Intermezzo 4: The Jericho Commandments and the COBIT processes

An interesting idea would be the mapping between the COBIT Control Objectives (CO) and the Jericho Forum Commandments (as defined in appendix A1). In Table 1 a first attempt is made. It is interesting to see the JFC's cannot be mapped entirely to the COBIT COs.

JFC's	*COBIT Control Objectives*
1: The scope and level of protection should be specific and appropriate to the asset at risk.	PO2.1, PO2.3, PO7.3, DS4.1,2,3,4,9, (PO9 according to (Forum 2007d))
2: Security mechanisms must be pervasive, simple, scalable and easy to manage.	(OPT PO2.1, PO2.3, PO7.3, PO4.8, PO5.X, PO8.1, po9.x, AI3.2, AI6.X, AI7.X, AI3.x, AI4.x, AI5.x, DS3.x) DS5.X (OPT DS9.X, DS12.X, ME4.X)
3: Assume context at your peril.	PO3.1-3, (OPT. PO4.5), PO9.1-5, (OPT.PO10.9), AI1.2 (opt. AI1.x), DS5.5-7,9,10 (OPT. DS 7.X, DS8.2,5, DS10.X, DS 12.X, DS13.3)
4: Devices and applications must communicate using open, secure protocols.	PO3.4, PO8.3, (OPT. AI.2.4, DS 12.X)
5: All devices must be capable of maintaining their security policy on an un-trusted network.	PO 6.3 (OPT. D12?)
6: All people, processes, technology must have declared and transparent levels of trust for any transaction to take place.	(OPT. PO7.2, 3, 5, 6, 7, PO10.4, 10.8, AI3.2, AI4.1,2, AI5.1,2, DS3.5, DS5.3,4, 5.11, ME2.X ME3.X, ME4.7)
7: Mutual trust assurance levels must be determinable.	(OPT.PO4.X, DS1.X, DS2.X, DS5.5, 5.11, ME2.1-7, ME3.X, ME4.7)
8: Authentication, authorisation and accountability must interoperate / exchange outside of your locus / area of control.	(DS 5.3 according to (Forum 2007d)
9: Access to data should be controlled by security attributes of the data itself.	
10: Data privacy (and security of any asset of sufficiently high value) requires segregation of duties/privileges.	PO4.9 (OPT: PO4.X-roles includen?), PO4.11, DS5.7 (OPT: DS11.2,3,6, DS12.3, DS13.4)
11: By default, data must be appropriately secured when stored, in transit and in use.	PO2.1-4 (OPT: AI3. DS4.1,2,3,4,9, DS5.8), DS 5.11 (OPT. DS11.x, DS12.2,3, DS13.4)

Table 1: Mapping between JFC's and COBITs Control Objectives

We will see more of this intermezzo in section 2.6.10 where we will look at IT-auditing in a de-perimeterised environment.
The control objectives between ( ) are optional and could be (often indirectly) related to the

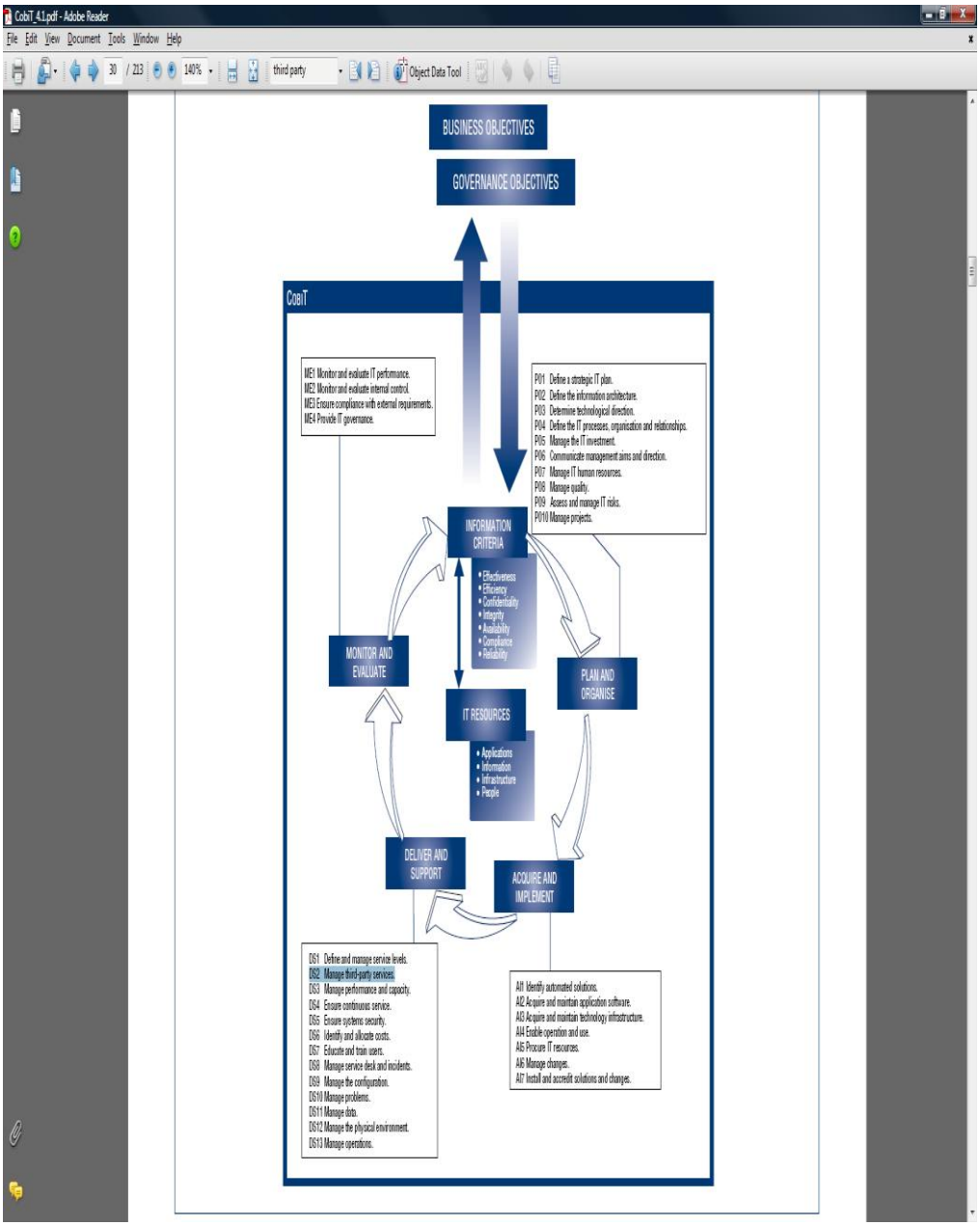


Figure 9: Overall COBIT Framework. (Institute 2007a)

2.5.Collaboration

2.5.1.Introduction

The Collaboration Oriented Architecture Framework is designed to provide a set of measures to collaborate securely with other individuals and complete organisations (see chapter 3 for more details). Yet, in order to fully understand the value of the COA framework and to be capable of using it in its full potential, one will need to know more about collaboration itself. That is why collaboration and many of the theories surrounding the subject are a substantial part of the basics (See Figure 3). We have to study different topics in the field of collaboration, such as:

- The definition of collaboration, which is studied in section 2.5.2 in relation to the concept of COA.
- The benefits and importance of collaboration, which is studied in section 2.5.3.
- A few important roles and relations one will find in collaboration, which is defined in section 2.5.4.
- Some aspects of collaboration in the perspective of the Prisoners' Dilemma (section 2.5.5).
- Collaborative Internet business models are briefly studied in section 2.5.6.
- The collaborative landscape in which many types of collaborative relations are described, is briefly addressed in section 2.5.7.

We will finish our study on the field of collaboration in section 2.5.8 by giving a short summary of what was studied and what aspects will be used as a focus in chapter 3 and 4.

2.5.2.What is collaboration?

The Cambridge Dictionary defines collaboration as:

> *"when two or more people work together to create or achieve the same thing".*
>
> (Cambridge 2007)

Kasten and Welborn define it as:

> *"Two or more entities working together to create mutual value".*
>
> (Ralph Welborn 2008)

The Collaboration in the acronym COA stands for exactly that: a COA has the capability of securely working together with other COAs towards a common or shared goal. The COA framework supports different ways of doing so: from sharing services to exchanging information. All can be done, irrespectively of time and location of the parties, in a secure way.
Mintzberg et al. gave a good overview of different types of collaboration (see also Figure 10). Here, the following types of collaboration were identified:

- **Intra-organisational:** collaboration between people inside the organisation such as working teams, task forces and committees, as well as collaboration across business units in the same diversified company.
- **Inter-organisational:** a type of collaboration between different organisations. Mintzberg defined a few directions of inter-organisational collaboration:
  - *Upstream:* collaboration with suppliers and other parties upwards in the supply chain.
  - *Downstream:* collaboration with franchisees and other parties downwards the supply chain.
  - *Governmental:* collaboration with the government.
  - *Lateral:* collaboration in shared research projects.

  (Henry Mintzberg 1996)

Mintzberg also defined a set of "how's" in which he defined certain forms of collaboration such as an ownership, licensing, joint ventures, alliances and networks. Furthermore, he defined a set of behaviours like negotiation, teamwork, networking, et cetera. (Henry Mintzberg 1996)
In (Chaffey 2004) we found multiple categories for collaboration, such as business-to-business (a joint venture), business-to-consumer (by using a customer forum like Dell does), government-to-business (various outsourced ICT projects), et cetera.

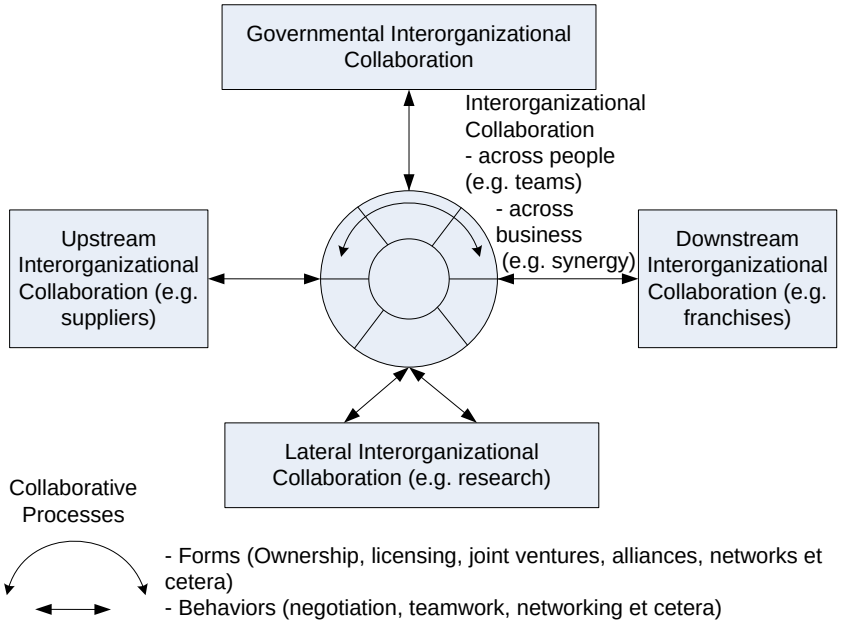


Figure 10: A model of Collaboration types, based on (Henry Mintzberg 1996).

There are many ways to describe collaboration, such as the dynamic networks of organisational entities that can work together, align their strategies and define their own relation and appropriate network strategy (GODFROIJ 1981a).

Another theory that looks like the dynamic network theory is the approach of the de-constructed firm by Joziasse in (Joziasse 2008): a deconstructed firm is an organisation that is focused on a subset of value-adding activities, traditionally performed in one integrated organisation. In order to create the same value as an integrated firm, a deconstructed firm has to collaborate with other deconstructed firms by use of an intimate partnership. A good example is the Dutch company NS (Nederlandse Spoorwegen), which split up into many smaller firms. Collaboration from this perspective is based on adding value to one's products or services. An intimate partnership is used as a link between the two parties in order to create a type of value chain that will deliver the product or service as requested. Important concepts in this context are business-, direct- and indirect relationships and interdependence, see (Joziasse 2008) for more details.

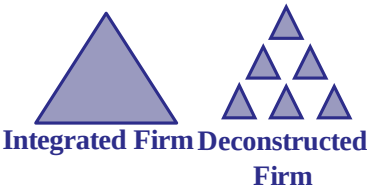


Figure 11: Deconstructed firm versus integrated firm. (Joziasse 2008)

As we have seen until now that collaboration can be used to achieve anything: it is a method increasing capacities in order to achieve a certain goal. This can vary from adding value to a product, or creating a product altogether, to creating a complete organisation based on different smaller organisations that focus on their own core-competencies.

One can also look at collaboration from a social action systems point of view, where there are multiple strategies an actor in such a system can use. The usage of the different strategies results in different game types (GODFROIJ 1981b). Even though the description is out of the

focus of this thesis, it does show that, based on how the collaborators act, different collaboration processes will be started. [54]

More characteristics of collaboration follow in this paragraph, based on the different roles and relations that have been defined by Joziasse in (Joziasse 2008) (section 2.5.4); the perspective of the Prisoners' Dilemma (section 2.5.5) and the collaborative landscape (section 2.5.7).
An important thing to remember is what collaboration as a principle is about – achieving a common or shared goal – and we should understand by now that collaboration is not a static process. It is an interaction between parties that is not always the same. Furthermore, one should understand why a COA is so important from this perspective; it helps the involved parties to collaborate securely.

2.5.3. Benefits and importance of Collaboration

In order to understand the benefits and the importance of collaboration, one will first have to look at its triggers. That is because the importance of collaboration is found in its cause. There is a variety of interrelated causes, as summarised by (Joziasse 2008): globalization, deregulation, the advent of developing markets like Brazil, Russia, India and China, increased outsourcing, augmented influence of shareholders, technical developments, increased complexity (of technology and management), deregulation and privatisation and the improved reach and usage of digital communication (Joziasse 2008). There are more reasons, as one can read in the introduction of this thesis.

Collaboration becomes an absolute necessity in order to survive on the market, with its fast innovation, it becomes important to focus on core competence. If one wants to deliver high quality and innovative products, then many aspects should be considered. It is almost impossible to do that by oneself. By using third parties' core competence as an addition to one's value, an improved and greater value can be brought to the market with all its obvious effects. (Boonstra 2002; Chaffey 2004)
When companies start doing so it becomes important to understand what impact it will have on one's business, once a collaborative partner stops putting in its collaborative value. (Ralph Welborn 2008).
There are many benefits to collaboration, such as:

- **Better service:** Better service to clients either by collaborating with them (for better feedback) or by other qualities: the increasing capabilities reached by a partnership, the resources shared by the partners (e.g. knowledge about clients and other third parties)
- **Economic realities:** increased efficiency, lower costs, et cetera. This can be achieved by focussing on core competence and/or increasing (production) scale. This also counts for innovation: by sharing the costs of innovation, one is able to achieve a more efficient innovation. Another field is the better competitiveness that is achieved by using networks.
- **Lower complexity:** Collaboration allows one to distribute tasks of a very complex process, so that each party can perform a combination of easier tasks instead of the complete complex process.
- **Resource sharing:** One can share knowledge, skills and physical assets.
- **Knowledge channels:** collaborative linkages can provide access to knowledge channels, serving as information conduits through which news of technical breakthroughs, new insights to problems or failed approaches travel from one firm to another.
- **Respond to a crisis:** If one faces a crisis, collaboration can help expand one's resources, giving him a better capability of facing that crisis.

[54] We could continue to describe different ways of collaboration such as in the virtual corporation, but most of those descriptions are out of the focus of this thesis.

- **Others:** There are other benefits, such as reducing uncertainty, fast acquisition of resources, enhanced legitimacy and the faster attainment of collective goals.

(Ahuja 2000; Mattessich, Murray-Close et al. 2001; Joziasse 2008)

Even though there are many benefits, the amount of benefits one can truly have from the collaboration is depending on the type of collaboration and the amount and type of relationships. See also (Ahuja 2000) because it is out of the scope of this thesis.

2.5.4. Roles and relations in collaboration

Joziasse provides some really important views in (Joziasse 2008) around collaborative networks or de-constructed firms, which show some of the complexity around the collaborative relationships of today. These insights help us to understand which factors have to be taken into account when one wants to steer a collaborative relationship, or apply and/or enforce certain rules and regulations in a collaborative relationship. They will actually influence the formation of the COA framework components and the way those components may be used by the organisations.

Other stakeholders:
First, it is needed to understand that, if one wants to look at a collaborative relation between two or more parties, one will also have to look on the parties that influence the collaborative relation. These are known as stakeholders. They are defined as "any group or individual that can affect or is affected by the achievement or of the firm's objectives". These groups or individuals can be anything, varying from shareholders to family, from clients to suppliers, from the government to interest groups. An important trend is the fact that the influence of some of the stakeholder groups, such as the government and the civil society has increased over the years. For more details, see (Joziasse 2008).

Primary and secondary relations
One should notice that there are two types of relationships. Primary direct dyadic relationships between two parties (see I in Figure 12), which directly influence the activities, actors, resources and ideas, of both parties. Secondary indirect relationships show the influence that multiple relationships have on each other (see II in Figure 12). Secondary functions of business relationships occur when resources are controlled by more than two organisations, when activities stretch more than two organisations and when perceptions are shared between more than two organisations. See Joziasse for more details (Joziasse 2008).

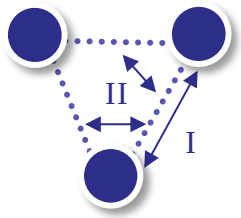


Figure 12: Different relations. (Joziasse 2008)

Roles in a network:
The inter-organisational relations often form a network of connected relationships between different actors. In this network, each actor takes up a role. This role is not only played in the network of collaborative parties or actors, it is also as a response to all stakeholders in that network.

		Centrality of the focal organization: *High*	Centrality of the focal organization: *Low*
Density of the Stakeholder Network	*High*	Compromiser	Subordinate
	Low	Commander	Solitarian

Figure 13: Organisational responses to stakeholder pressures. (Joziasse 2008)

The role that is played, is based on multiple factors. Joziasse shows in (Joziasse 2008) that the role against the stakeholders is based on the factors:

- **Network Density:** *"the degree of interconnectedness of the whole network surrounding the focal organisation"* (Joziasse 2008).
- **Centrality of the focal firm: "***the degree in which the focal organisation has formal and informal power in the network"* (Joziasse 2008).

This leads to the following roles:

- **Compromiser**: tries to balance and bargain with its influential stakeholders in a dense network.
- **Commander**: can shape the behaviour expectations that stakeholders have of its organisation in a low interconnected network.
- **Subordinate**: is unable to control the information exchange from its sideline position in a dense network.
- **Solitarian**: does not really play a role in a network of collaborating parties, since it is always dependant on other actors to deliver a viable service or product.

(Joziasse 2008)

Seeing these different roles, it should not be too hard to understand that some organisations are allowed to dictate more rules to their partners than others. A commander is able to have more influence in the network than subordinates for instance. This partially explains why the concepts of SOA and SAAS as full collaborative partners did not work entirely. The business- and technology-based approaches often assume equal roles between companies, which is not the case. In fact, some of the governance and IT Service Management frameworks assume either this equality or assume nothing at all and try to bypass it.
If one wants to establish a secure environment for dynamic collaborations, these roles should certainly be kept in mind.
The exact details around the roles are out of the scope of this thesis, see for more information (Joziasse 2008).

Adding more complexity: complementary and competitive relationships:
To add a little complexity, which is required to understand more about collaboration, we must look at another aspect that was also described by Joziasse in (Joziasse 2008). The aspect of the competitiveness and complementarity of relationships:

- **Competitiveness of relationships:** This gives the organisation the ability to choose and hence strengthen the bargaining position in the network. It adds a degree of flexibility since there is less reliance on a limited number of relationships. It requires a mutual adaptability from both sides. (Joziasse 2008)
- Complementarity of relationships: This *"gives organisations the ability to create competencies as a network. This inherently increases the interdependence in the network. The complementarity of resource ties can have a beneficial effect on the resource value, its effectiveness and efficiency. The complementarity of activities is the degree in which activities are part of the same activity chain. A complementary activity chain can aid in the overall network efficacy. The complementarity of schema configurations signifies the ways in which the viewpoints of different actors together create a more holistic view of their strategic context."* (Joziasse 2008)

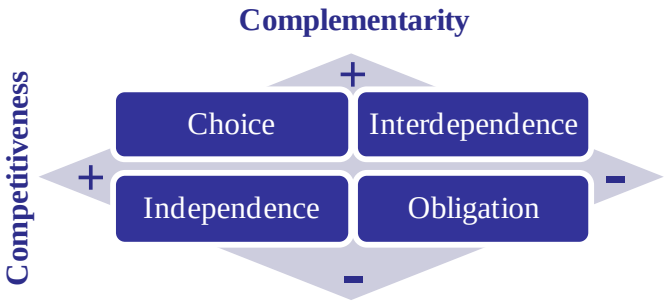


Figure 14: Strategic meaning of the relationship. (Joziasse 2008)

The two characterizations of relationships allow us to see if a company has any freedom in choosing a collaborative partner by determining the strategic meaning of the relationship (see Figure 14):

- **Choice:** In case a relationship structure has both a high competitiveness and high complementarity towards other relationships, it offers the organisation a choice. The complementary character of the relationships make it important to the network, yet, since there is also room for competition in this value added part, this leaves the deconstructed firm a choice.
- **Interdependence:** This can either happen due to low complementarity or due to low competitiveness:
  - *Low complementarity/high competitiveness:* It is difficult to choose a different configuration of relationships; either the organisation has strategic reasons to stay committed to the current configuration or there simply are few potential actors that focus on delivering the type of value added service. This means that the organisation will face a high dependence with regard to that structure of relationships.
  - *Low competitiveness/high complementarity:* It is an important constellation of relationships, yet there is little room in the organisations' current relationships to choose differently. It has different reasons. This means the organisation faces a high dependence with regard to that structure of relationships.
- **Obligation:** If there are no complementary or competitive relations, yet collaboration with such a party is a necessity, it becomes an obligation. These are the limiting relationships.

(Joziasse 2008)

Seeing the different strategic meanings of relationships, it should not be too hard to understand some relationships are allowed to dictate more to the focal organisation than others. For instance, an obligatory relationship is allowed to dictate or limit the organisation more than a choice relationship. If the focal organisation does not like the limits imposed by the chosen relationship, it can opt for another relationship that is less demanding, or demands otherwise. If the focal organisation does not like the demands imposed by an obligatory relationship, the focal organisation might have to give in, since there is no alternative.

Adding even more complexity: multiple roles in multiple networks:
Another complexity-adding observation from Joziasse is the fact that most of the collaborating organisations are a part of multiple networks. This means they are automatically playing multiple roles. Although an organisation can be a commander in one network, it may be subordinate in another. Knowing this, it means that an organisation that is allowed to dictate rules in a certain network, might be obliged to follow the rules in another. In fact, two actors that might be in the same two or more networks could be changing positions: the commander from network X can be a subordinate in network Y and the subordinate in network X can be the commander in network Y. Thus, there is no static way of dictating, or creation of rules and regulation for a collaborative relationship.
This implies there will be a need for better governance between the collaborating organisations and, inherently, within those organisations. Due to this complexity, it is inevitable there should be some kind of toolset to manage these different roles with and make them transparent to the management. However, it is not yet clear if the COA framework is the framework to add such a toolset to (see chapter 3 for more details).

Some remarks around strategic control:
Before summarizing and concluding this section, there is one issue to address: the control inside a network. Joziasse said the following:

> *"Conventional, stringent control in such an interrelated context is suboptimal and even counterproductive since monitoring, restraining and directing leaves no room for network actors to organise their business independently. Nor is a state of authority of one actor over the other beneficial to the overall network effectiveness since it also*

restricts or directs these others. Thus control should move from centrally held to a more decentralized, empowered approach for network actors."

(Joziasse 2008)

This means one will have to find new means of control in a collaborative relationship or, better said, in the network of collaborative relationships.
Joziasse also stated in that same piece that a network organisation or deconstructed firm is part of a network. This network itself has a purpose to which every actor in the network contributes in its own way. Thus, the network purpose will influence the operational choices and actions of actors in the network, as well as the inherent interaction. At the other hand, other factors, such as Political, Economical, Social or Technological developments will affect the purpose and constitution of the network.

Concluding, and what's more:
Looking back in this section, we see that the study of Joziasse gives us many insights in collaborative networks and relationships:

- **Stakeholders:** First, we have seen that other stakeholders, besides the party which one wants to collaborate with, have to be kept in mind.
- **Primary and secondary relations:** Second, we have seen that we should not just look at only the primary relationships, yet also at the secondary ones that influence the parties we collaborate with, as well as on the focal organisation itself.
- **Roles in a network:** Third, we have looked at the roles in a network: there are different roles in a network that allow players to have different functions and possibilities for control inside the network. For instance, a commander is allowed to shape the behavioural expectations of stakeholders, while a subordinate is unable to control the information exchange at all!
- **Strategic meaning of relationships:** Fourth, we saw the strategic meaning of relationships, which can be derived from the competitiveness and the complementarity of the relationships. It has influence on the amount of control a player might have in a network.
- **Multiple networks, give one actor multiple roles:** Fifth, we have seen that actors can be part of multiple networks in which they can perform different roles. An actor can be a commander in one network, yet a subordinate in another.
- **Strategic control has to change:** Last, we saw that the traditional way of conventional, stringent control will not work in a collaborative relationship and that the network purpose offers more possibilities and insights.

Even though we have learned from these insights, many are still left undiscussed around the characteristics of collaborative relationships and networks. Most of this is out of the scope of this thesis. See (Joziasse 2008) for more details.

2.5.5. Collaboration and Prisoners' Dilemma

Another view on collaboration is defined by the Prisoners' Dilemma, which is part of Game Theory. It can be summarised as follows: Two prisoners are asked to testify against each other. If they do, they gain freedom. If both testify, they both get an intermediate sentence, if one testifies and the other stays silent he will be set free and the other will get a long sentence. If both stay silent, they both get a short sentence. (Demarteau 2008)

If we look at collaboration from the perspective of the Prisoners' Dilemma, then we see that both of the collaborating parties will have to sacrifice something in order to gain from the collaboration. If however one sacrifices and the other does not, an unbalanced profit will most certainly be the case: one will have the maximum profit and the other will have a great loss.

So how can the collaborating parties be sure of the other that they all are willing to sacrifice? Some sort of assurance is needed. This assurance is found in trust. If both parties can trust each other to sacrifice, a safe collaboration based on mutual benefit will occur.

More details about trust are found in section 2.6.8 and in (Demarteau 2008; Leijden 2008). It suffices to realise for now that collaboration from this perspective is basically about investing a little into the collaborative relationship and trusting the other party to do so as well.

2.5.6. Internet business models

As said, there are multiple forms of collaboration: from sharing a single service to the exchange of information, from a single transaction to sharing complete processes. Knowing there are so many forms of collaboration, there is no way to exactly specify every single one of them. There is a group of forms already specified, which we can use from the Jericho perspective: the Internet Business Models as defined by prof. Michael Rappa[55]. These models can show us how the users of the internet already instated a set of collaborative models. These models are already studied in the Jericho perspective, so they should give us extra data about what to expect of a COA. We list the models here, add a few examples and look at the models from a Jericho perspective:

- **Brokerage:** Brokers are market makers; they bring buyers and sellers together to facilitate transactions. They operate in multiple markets like business-to-business, business-to-consumer. There are different brokerage models, such as distributor, search agent, virtual market place, transaction broker, auction broker, service hubs and a marketplace exchange. Most of these models can be important from a Jericho perspective. For instance within the trust broker (see section 2.6.8 for more details), a sort of transaction broker is used. Besides the direct use of the brokerage models within the Jericho model, most of these brokerage models are also interesting in that information is exchanged between the broker and two or more parties. This automatically means that some form of (secure) interconnection is necessary. If we look at the mentioned models we see that they are all about a certain form of collaboration, varying from a single transaction (transaction brokers) to complete processes (search agents).[56]
- **Advertising:** Advertising is an extension to the traditional media broadcast model. The broadcaster, for instance a web site, provides content (usually, but not necessarily, for free) and services (like email, IM, blogs) mixed with advertising messages in the form of banners. The banner ads may be the major or sole source of revenue for the broadcaster. There are different advertising models such as portals, classifieds, user registration, query-based paid placement, contextual advertising/behavioural marketing, content-targeted advertising, infomercials and ultramercials. The portals are interesting from a Jericho perspective, because if we look at the modern portals of today (Yahoo, iGoogle, et cetera) we see some of them can be customised to the personal needs of the customer. What's even more important, many online services can be added to those portals. Take iGoogle for instance: it is possible to aggregate the output from many third parties into the portal. With that, it is not only a portal, it is broker as well. The portal of today allows you to exchange information with third parties, making it a new type of broker and object for collaboration. Another interesting form is the user registration: one is able to enter free content only after registering information. This information can be used for multiple purposes such as marketing or analysis of a certain group. It may even be an extra opportunity to present certain objects that are (more specifically) directed at that specific user. The user registration is interesting from a Jericho perspective because a lot of information exchange occurs. This means there

[55] See also http://digitalenterprise.org/models/models.html , visited at 1-09-08.

[56] There are also brokers that connect multiple services like a SaaS or SOA hub.

will be a lot of interconnection and thus, that there is a demand for correct and trustworthy information (-exchange).

- **Infomediary:** An infomediary (Information Intermediary) is a type of data aggregator mostly focused on either consumers and their behaviour (demographic, statistics, (surfing) behaviour, et cetera) or producers and their products (reviews, statistics, work, process et cetera). They help either producers or consumers to understand any given market. There are different types of infomediaries such as advertising networks, audience measurement services, incentive marketing and a metamediary. Many of these types crucial from a Jericho perspective. A good example is the advertising networks. They consist of a series of feed banner ads to a network of member sites. These ad networks collect data about web users that can be used to analyze marketing effectiveness and visitor statistics. This information can be very useful for multiple parties, such as the party that hosts the sites the banners are displayed on, as well as the marketers and the designers of the banners. This information can be spread or sold by the organisation that manages or contracts the advertising networks. This makes it interesting from the Jericho point of view because of the necessary collaboration and interconnection processes.
- **Merchant:** A merchant is a digital version of the already known merchants: wholesalers and retailers of goods and services. There are different forms of the merchant model: virtual merchant, catalogue merchant, click and mortar and a bit vendor. The merchant model is interesting in the fact that it is focussed on transaction-based collaboration, which creates (long- or short-term) financial relations. This form of collaboration is again based on trust and requires a secure form of interconnection and information exchange.
- **Manufacturer:** This model is also called the “direct model”. It allows the manufacturer to reach buyers directly, thereby bypassing (other) merchants. The manufacturer model can be based on efficiency, improved customer service, and a better understanding of customer preferences. There are different forms such as purchase, lease, license and brand-integrated content. One of the interesting things about the manufacturer model from a Jericho point of view is that a manufacturer must interconnect and collaborate with many more different parties in comparison to a retail- or wholesale-based sale. Many customers, transport, sales, third parties, all of these parties will be connected to and collaborate with the manufacturer.
- **Affiliate:** The affiliate model provides purchase opportunities wherever people may be surfing. It does this by offering financial incentives (in the form of a percentage of revenue) to affiliated partner sites. The affiliates provide purchase-point click-through to the merchant. It is a pay-for-performance model: if an affiliate does not generate sales, it represents no cost to the merchant. Variations include banner exchange, pay-per-click, and revenue sharing programs. This model is vital from the Jericho perspective, being based on interconnection and collaboration between different organisations, web sites and systems.
- **Community:** The community model consists of different types of virtual communities that are based on user loyalty. Users have a high investment in both time and emotion. Revenue can be based on the sale of ancillary products and services or voluntary contributions. Revenue may also be tied to contextual advertising and subscriptions for premium services. There are different types of the community model: open source, open content, public broadcasting and social networking services. All of them are interesting in the fact that they increase the demand for interconnection. The community models get their value strictly from collaboration between the different users. Collaboration is the main value-adding process for these communities.
- **Subscription:** The subscription model is based on a periodic fee to subscribe to a service. There are different types of the subscription model, such as content services, person-to-person networking services, trust services, internet service providers. Both the trust services and the person-to-person networking services are in this picture since they deal with various forms of trust, interconnection and collaboration. More about the Trust Services is discussed in section 2.6.8.
- **Utility:** The utility model is also called the “on-demand” model. It is based on metering usage, or a "pay as you go" approach. Unlike subscriber services, metered services are based on

actual usage rates. There are different types of the utility model like metered usage and metered subscription. Both are not really interesting in the scope of this thesis, besides the fact that one will have to measure the actual usage of the data or the service, which however is also out of scope here.

See also (Marle 2007) for more details about the security around these models and the online reference as for further explanation of the types that have been mentioned. Both of the subjects are out of the scope of this thesis.
For an understanding of the Jericho approach, concepts and ideas: see the next paragraph.

2.5.7. The collaborative landscape

Another way to look at collaboration is in the view of the collaborative landscape, as described in 'The Jericho principle' by Ralph Welborn and Vince Kasten. This creates an easy way to map different ways of collaboration to certain measures. They use the following measures:

- **Intimacy***:* Is a measure of the degree to which participants of a collaborative effort expose their core competencies and value to one another.
- **Dynamism**: Is a measure of the length of time the collaboration is expected to last.

These two measures can be used to create a map with four zones:

Figure 15: The collaborative landscape. (Ralph Welborn 2008)

- **Country clubs**: There is a stable set of members who know one another socially, but not necessarily intimately. The relation is stable yet less intimate.
- **Bars**: It is a quadrant of highly dynamic relationships that do not last long and are not very intimate.
- **Commitment**: This quadrant is one of very stable, highly intimate collaborations.
- **Jericho**: This is the zone where highly intimate collaborations take place, but with less stable relations. Being in the Jericho zone means that each partner has de-perimeterised, quantifies the value of each partner to a relation, controls the risk with the high intimacy, and equitably shares the rewards of the collaboration. This is the zone where the parties can be found that want to implement the COA framework and utilise it to the max.

(Ralph Welborn 2008)

One can see, based on these criteria, where his collaborative relationships can be mapped to, and what this means for him and the use of a COA.

2.5.8. Concluding: Collaboration and the focus for this thesis

Looking back on this paragraph, we conclude the following:

- **Collaboration:** Is in principle about working together with one or more entities to achieve a common or shared goal. There are different types of collaboration: intra-organisational is a type of collaboration inside the organisations, while inter-organisational collaboration happens between two organisations. There are multiple ways to collaborate: by negotiation, teamwork, networking, et cetera.

- **Views on collaboration:** There are multiple views to define collaboration such as the dynamic network theory, social action system theory or the deconstructed firm theory.
- **Benefits and importance of collaboration:** Collaboration is very important: in order to survive on the market it becomes important to focus on core competence and to use the core competence of third parties to deliver an improved value to the market. There are multiple benefits, such as better service, economic realities (efficiency, scale, et cetera), lower complexity by dividing the problem between parties, resource sharing, knowledge channels, better response to a crisis and many others.
- **Stakeholders will have influence:** Besides the parties that collaborate, there are many other entities that influence the collaboration and the parties involved. These are the stakeholders.
- **Primary and secondary relations:** We have seen that we should not only look at the direct primary relationships, but also at the secondary, which have influence on the parties we collaborate with and on the focal organisation itself.
- **Roles in a network:** There are different roles in a network that allow players to have different roles and possibilities for control inside that network: for instance, a commander is allowed to shape the behavioural expectations of stakeholders, while a subordinate is unable to control the information exchange at all!
- **Strategic meaning of relationships:** The strategic meaning of relationships, which derive from the competitiveness and the complementarity of the relationships, influences the amount of control that a player may or may not have in a network.
- **Multiple networks give one actor multiple roles:** The actors can be part of multiple networks in which they can perform different roles. An actor can be a commander in one network, yet a subordinate in another.
- **Strategic control has to change:** The traditional way of conventional, stringent control will not work in a collaborative relationship but the network purpose offers more possibilities and insights.
- **The Prisoners' Dilemma:** The Prisoners' Dilemma tells us that, if both parties of a collaborative relationship want to benefit, they both have to sacrifice a little. In order to be sure that both parties will do so, trust is a necessity.
- **Internet business models:** There are various internet business models that use a certain form of collaboration such as brokerage, advertising, infomediary, merchant, manufacturer, affiliate, community and subscription. Most of these forms are interesting because a certain form of collaboration is a necessity and that certain ideas and concepts of the Jericho Forum will be necessary as well.
- **The collaborative landscape:** The collaborative landscape shows different styles of collaborative relationships based on intimacy and dynamism. It defines the types of collaboration: country clubs, bars, commitment and Jericho. The Jericho type takes maximum benefit from a COA framework.

So what do we focus on for this thesis? All of the information above is used in detailing the COA framework in this study. Furthermore, the aspects of roles and relations can be used to understand how two or more COAs should interact.
Looking back at this paragraph, we can see that we have just covered only a small part of the theories around collaboration. In order to fully understand the concepts behind a collaborative relationship, the author recommends the literature used to write this paragraph.

2.6.The Jericho Forum and its concepts

2.6.1.Introduction and overview

In this section we look closely at the concepts of the Jericho Forum. These concepts will become the cornerstone for the COA framework and need to be understood before we can describe the COA framework. (See also Figure 3)
It is important to notice there are many studies carried out by my peers at the Security and Innovation Research Centre considering these topics. This is why in certain fields, instead of reinventing the wheel we limit to using some of the results of those studies to describe the ideas. For more details regarding each subject presented here, the reference list containing the indicated materials should be consulted.

Jericho Core Components
In "Authentication and accounting" by Evgeny Barannikov, a set of Jerihcos Core Components are explained in relation to each other. The author of this thesis recommends reading this work in order to find a better understanding of the Jericho Concepts all together.

This section consists of the elements:

- The paragraph, lined out in section 2.6.1.
- The Jericho Forum, described in section 2.6.2.
- The central theme of the Jericho Forum: 'De-perimeterisation' is discussed in section 2.6.3.
- Other concepts, ideas and measures that are needed for this de-perimeterisation will be discussed in the remaining sections of this paragraph. Note that all of these concepts reoccur in chapter 0, whereas almost every concept of the Jericho Forum will be necessary to build a functioning Collaboration Oriented Architecture.
  - The use of inherently secure protocols, open standards and systems that are "secure out of the box" is discussed in section 2.6.4.
  - The use of wireless in a de-perimeterised environment is discussed in section 2.6.4 as an intermezzo.
  - The use of policy management is discussed in section 2.6.5.
  - The use of Data classification and data privacy is discussed in section 2.6.6.
  - The use of Identity management, user authentication and federation is discussed in section 2.6.7.
  - The issues around trust and the usage of trust, trust brokers and trust management, is discussed in section 2.6.8.
  - The use of end-point security is discussed in section 2.6.9.
  - The use of IT-Audit in a de-perimeterised environment is discussed in section 2.6.10.
  - The focus of this thesis and a little summary considering the themes of the Jericho Forum will be given in section 2.6.11.

Jericho and the Collaboration Oriented Architecture
in order to enable network based de-perimeterisation and, consequently, collaboration, the Jericho Forum created the Collaboration Oriented Architecture (COA). That is why every concept mentioned in paragraph 2.6 can be mapped to parts of the COA in chapter 0.
This mapping can be found in section 3.10.

Sections 2.6.2 till 2.6.10 will have an additional "COA V2.0" box added to them, in which all new requirements and additional information will be summarised based on the new V2.0 release of the COA framework.

2.6.2.The Jericho Forum

The concept of Jericho is based on a story from the Holy Bible:
There was a city in the south east, now known as Palestine, called Jericho. The city relied on its famous thick walls and even the people of Israel did not think they could overtake that city easily. Later on, the unthinkable happened. The walls came falling down and the people of Jericho, relying on their defensive walls that just fell, were not capable of stopping the Israelites. Almost every man inside the city fell by the sword that day. (Bible 4000 BC)

The story of Jericho is analogue to how many organisations of today handle their data: they feel safe, putting up firewalls all around their perimeter and think that nothing problematic can

happen. The problem is that firewalls do fail lately. There are often holes in the perimeter security when technologies such as VPN and other Collaboration tools are implemented.

Since the summer of 2003, there is a group of CISOs that have a common interest in this topic. This led to the founding of a forum in January 2004[57], by the Open Group[58].

The Open Group describes the main goal of the Jericho Forum:

> *"The huge explosion in business use of the Web protocols means that today the traditional "firewalled" approach to securing a network boundary is at best flawed, and at worst ineffective. Examples include:*
>
> - *business demands that tunnel through perimeters or bypass them altogether*
>   - *IT products that cross the boundary, encapsulating their protocols within Web protocols*
>   - *Security exploits that use e-mail and Web to get through the perimeter.*
> - *to respond to future business needs, the break-down of the traditional distinctions between "your" network and "ours" is inevitable*
> - *Increasingly, information will flow between business organisations over shared and third-party-networks, so that ultimately the only reliable security strategy is to protect the information itself, rather than the network and the rest of the IT infrastructure.*
>
> *This trend is what we call "de-perimeterisation". It has been developing for several years now. We believe it must be central to all IT security strategies today."*
>
> (Forum 2008c)

The Forum has set up a vision and a mission in 2006:

Vision statement:
To enable business confidence for collaboration and commerce beyond the constraint of the corporate, government, academic and home office perimeters, principally through:

- Cross-organisational security processes and services;
- ICT products that conform to open security standards;
- Assurance processes that, when used in one organisation, can be trusted by others.

Mission statement:
Act as a catalyst to accelerate the achievement of the collective vision, by:

- Defining the problem space;
- Communicating the collective vision;
- Challenging constraints and creating an environment for innovation;
- Demonstrating the market;
- Influencing future products and standards.

The vision statement should last three to five years while the mission statement remains constant. (Forum 2005)
The Jericho Forum released a set of 11 commandments that are the fundamental principles for dealing with de-perimeterisation. (See Appendix A for an outline)

[57] Members include Capgemini, HP, IBM, Sun Microsystems, NASA, Cisco Systems and MITRE. See www.theopengroup.org/jericho/faq-bo.htm, accessed on: 17 March 2008

[58] www.opgenroup.org, accessed on: 17 March 2008

2.6.3. De-perimeterisation explained

Introduction
The central theme of the Jericho Forum is de-perimeterisation. So what is that exactly?
De-perimeterisation is a word that first showed up in a paper from Jon Measham in 2001 with the title "*Value Less Security- can a relativistic approach to risk assessment lead to an extension of the Protect, Detect, React paradigm?*"
In that paper, he showed it is important to bear in mind what the exact value asset is for the owner of an object, and what the value asset is for an intruder. A great lesson was that these two are often different! This difference between the value assets can be used to improve the protection of the value assets of the owner. By focussing on the protection of those value assets that are important, boundaries, protection measures and their focus will shift towards the value assets of the owner. This is called de-perimeterisation. (Measham 2001)

Adding defence in depth

Whenever an organisation will de-perimeterise according to the concepts of the Jericho Forum, it will not throw away its firewall. The firewall is still very important as a macro perimeter. But the micro perimeters will have to be added .

So how does such a thing work? Before going through all the concepts and important ideas of the Jericho Forum, we observe a simple example from the physical world.

An Example: the ATM
In the old days, the technical designers tried to make an ATM itself secure (by a robust construction and firm mountings). Everybody thought this would be the way to secure an ATM: It would be harder to destruct the ATM and steal the cash cassette. The only thing they did not think about, was that the value asset to the intruder was different from the value asset of the Bank. The Bank wanted to make sure that they did not lose the money, but more important, they wanted to provide the customer with the service of cash withdrawal at an ATM at all times. The intruder simply wanted the money. Later on, the designers focussed more on the thing that really mattered: the cash cassette. If the cash cassette was protected by itself – by rendering the money useless as soon as the cassette was compromised - there would be no use of breaking into an ATM and thus the service would stay available. Without the reason of breaking into the ATM, all that had to be done was designing it firmly enough to withstand most of the vandalism, instead of making it an invulnerable fortress. This makes an ATM less expensive to build and gives it greater flexibility as to its location.
As one can see, the perimeter shifted: from the outside wall of the ATM to the inner core: the cash cassette. Thus, a perimeter has been redefined in order to create one that protects the assets that do matter. Later in this chapter, this example reoccurs in order to easily clarify a few things. (Measham 2001)

De-perimeterisation in short

De-perimeterisation is about securing and protecting the important assets with focussed and effective measures instead of protecting both the non- and important assets with the same measures.

Jericho and de-perimeterisation:
Now that we know what de-perimeterisation is, we can take a better look into the theme of the Jericho Forum:
De-perimeterisation as the Jericho Forum sees it, has four important components:

The first and most important is the shift in focus from network security to data-protection. This shift became a necessity because the members of the Forum understood that the current "fortress approach" to network security made it really hard for collaborating partners to interconnect and dynamically cooperate. The real value-asset that needs protection, is the data. The infrastructure is less important. This does not mean that it is not important at all, as we will see in the second component. De-perimeterisation is definitely not about throwing away firewalls or other defensive measures. They add to defence depth, which is still necessary. By using the network security, one can establish a layered approach to securing the most important asset: the information.

A good question would be: "How does this work?" The answer lies in the commandments the Jericho Forum released. These commandments[59] show which design principles are important for de-perimeterisation. On the field of data protection there are a few commandments that are elemental (see Appendix A for the line out of the commandments):

- JFC9 shows that data should be protected by itself, so there should be a perimeter around the data in its own source, by means of DRM or an alternative way of securing.
- JFC1 shows that the chosen protection measures should depend on the asset at risk. In other words, the more important the data, the heavier the protection measure.
  - JFC2 shows that the security mechanisms should be scalable and easy to manage, which means that certain coarse-grained levels of importance in order to classify the data is needed.[60]
- JFC11 shows that data must be appropriately secured when stored, in transit or in use: So the data should always be secured in an appropriate way. This also means that the amount of taken security measures (see also JFC1) will be variable. A little notice is given by the commandments V1.2: There may be data that does not need to be secured at all.
- JFC10 shows the importance of taking into account duties and privileges within the measures of protection. Besides protecting the data in general, there should be an independent control authority that manages permissions, keys, privileges et cetera.

(Forum 2007a)

Intermezzo 5: De-perimeterisation: the Cash Cassette and the Informational Asset rendered unavailable if compromised.

The analogy between the cash cassette and the informational asset goes further than may be expected:

As soon as the cash cassette is compromised, a smoke or a paint bomb blows inside the cassette, destroying all banknotes inside. The same thing should happen to information sources that are compromised: they should be rendered useless by the usage of DRM: either destroying the information or render it unavailable by any other means. (Measham 2001)

This set of commandments does not give a concrete answer of how de-perimeterisation should take place. They do however give a great starting point as a set of design principles that will be used in this work besides the set of Position Papers that are based on these commandments.

The second component is about how one should survive in the hostile world of the internet. Two JFCs are the centre of this component:

- The use of open, secure protocols (JFC4).
- The use of devices that are capable of maintaining their security policy on an untrusted network (JFC5).

(Forum 2007a)

This component will be investigated in section 2.6.4, 2.6.5 and 2.6.9 where we will look at different security mechanisms, services, protocols, policy management standards, recommendations and issues in this field.

The second component is about network and device security. This is, considering the layered approach, still of significant importance as well. The only difference is that the focus shifts from the perimeter to the usage of the correct secure protocols and devices that can handle the security policies and such protocols.

[59] At the time of writing this thesis, the current commandments are being called principles as well. There are members of the Jericho Forum who would like to see the commandments being referred to as principles.

[60] In order to classify the data to a certain level of importance, one should use a data classification system. See also section 2.6.6 for more information.

The third component is necessary to support the second: Identity, Management and federation. JFC8 summarises this component: *"Authentication, authorisation and accountability must interoperate/exchange outside of your locus/area of control"*. This will have a large impact on the perimeterised world of today. The component is discussed in section 2.6.7.

The forth component, which backs up the second and the third, is the need for trust and the usage of trust management. The basics of this concept lie in JFC6 and JFC7:

- All people, processes and technology must have declared, transparent levels of trust for any transaction to take place. (JFC6)
- Mutual trust assurance levels must be determinable.(JFC7)

(Forum 2007a)

To be capable of trusting each other, one has to be capable of ensuring the risks are mitigated and/or in control. In order to do so, IT-audit and End-point security become vital. Both topics will be addressed in this thesis (section 2.6.10 and 2.6.9)[61].

All of the components are backed up by the "basics", which we could see as "the fifth component" or "the fundamentals" as the Jericho Forum calls them. It is the first three JFCs:

- The scope and level of protection should be specific and appropriate to the asset at risk. (JFC1)
- Security mechanisms must be pervasive, simple, scalable and easy to manage. (JFC2)
- Assume context at your peril. (JFC3)

(Forum 2007a)

Most of the fundamentals can be found in any section, since they have an impact on each different section. Yet, there is one specific topic that stands out, in the light of JFC1, in importance: Data Classification. More details around that are found in section 2.6.6.

Reasons for de-perimeterisation:

So what are the reasons for de-perimeterisation? Why is it considered important? Here are a few:

- **No perfectly secure boundary**: It seems impossible to create the perfect network boundary that is 100% secure. So if one trusts only in the borders of the network, problems arise. (Stanton 2005)
- **The need for securing data itself**: Instead of securing all the network media, it is more logical to secure the data because that remains inherently secure during transit and storage.
- **More sophisticated and faster online threats and new kinds of attacks:** There is an ever increasing number of sophisticated and faster online threats. Network protection does simply not suffice.
- **Better business connectivity**: Business is demanding more connectivity outside the enterprise. It is hard in the current fortress approach to easily interconnect with other parties and collaborate securely. When de-perimeterised, one improves capability to interconnect and exchange information safely, irrespective of the location of collaborating members. This could lead to significant improvements such as:
  - *Better B-to-B integration*: Enable direct B-to-B integration of ERP systems with your major partners, enabling better exchange of data and closer co-operative working.
  - *Better border alignment*: Allow legal, commercial, and quality-of-service borders to align with the network and infrastructure implementation, paying only for the bandwidth and infrastructure the business actually needs.
  - *Better interaction with customers:* One can supply his customers with direct connections to different information sources and applications to improve interaction.

[61] Another interesting approach is the Jericho Security Architecture, which is out of the scope of this thesis, but still interesting enough to notice.

  - Increased usage of web services:
- **Cost reduction**: Accessing applications through a broadband-enabled device, using XML or Web Services, reduces the costs associated with connectivity and maintenance of leased lines, private exchanges and even VPNs.
- **Provide more flexibility**: One becomes capable of accessing the right applications and sources anytime and (from) anywhere. The flexibility is also enhanced by the fact that pervasive, fast, reliable and cheap Internet connectivity is becoming available everywhere.

(Forum 2005; Stanton 2005; Forum 2007f; Forum 2008e; Forum 2008c; Stan 2008a)

De-perimeterisation in practice:
Now how to deal with this? How can one protect his data/information and de-perimeterise? In order to answer this, the Jericho Forum released a set of Position Papers that describe many of the concepts, measurements and technologies that are necessary to be capable of de-perimeterising an organisation[62]. These are discussed in the following sections (2.6.4 to 2.6.10). All of these ideas can be found again in chapter 0 in the form of the Collaboration Oriented Architecture, where the answer to all questions is given in such a practical way that one can fully de-perimeterise its organisational borders.

Concluding: what is de-perimeterisation?
De-perimeterisation is about securing and protecting the important assets with focused and effective measures, instead of protecting both the invaluable and valuable assets with the same measures. Thus, no more "fortress-approach" for network security, but one needs to re-establish the perimeters around the information assets. [63]

A final note about the firewalls:
As described (Forum 2007c), the firewalls will not only add to defence depth, they will also be used as a quality of service (QoS) separator. This allows services and other sources to use different QoS specifications; whether the device or requester of the service is inside or outside the corporate environment. If he is inside, he will often have a lower latency and a higher bandwidth available towards the service than whenever he is outside of the corporate environment. As this is written, the Jericho Forum works on a new Position Paper around QoS (Jerichoforum 2007) in which they describe the necessity of Network QoS.
Other parties, such as (Demarteau 2008) argue that a firewall can also be used to separate sub networks.
One could argue the firewall can be used as an addition to the other measures of enforcing the Jericho Commandments. For instance, a firewall could filter out all traffic that is in violation with the fourth commandment: as soon as an insecure protocol is used, the firewall blocks that stream of data or warns that an insecure protocol is being used. See also the next section for more details.

2.6.4. Surviving in a hostile world by using open and inherently secure standards, systems, and protocols

Introduction
As pointed out in the previous section, there are many important subjects on the field of de-perimeterisation. One of them is the usage of inherently secure standards, systems and protocols. This traces back to the commandments: The fourth Jericho Commandment states that devices and applications must communicate using open, secure protocols.

[62] Besides the Position Papers, the Security Research and Innovation Centre at Capgemini created a few books based on these Position Papers. They will be used as a source as well.

[63] See Appendix A3 for a short overview of the roadmap to de-perimeterisation.

A good question then is: “Which protocols can we use? Which are secure?” The focus of this section will be the open and secure standards and protocols, in order to see which protocol we can surely use in a de-perimeterised environment and why. The following section is closely related to this one: In order to survive in a hostile world, one must be capable of maintaining the security policies even in a hostile environment. We will see more of that in the next section.

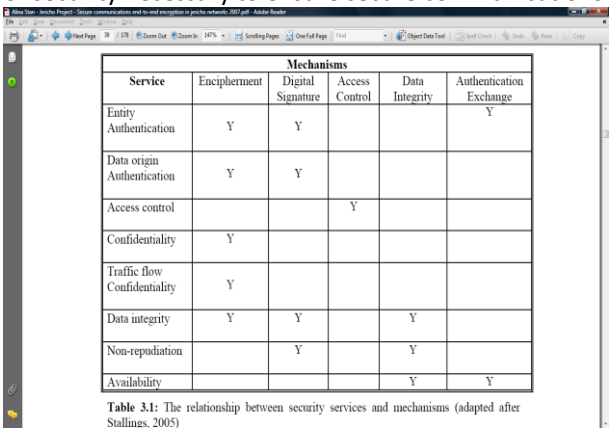
Getting back to the commandments: JFC4 states that devices and applications must communicate using open, secure protocols. Of course: the use of such should always be specific and appropriate to the asset at risk (JFC1). This implicates data classification has to take place before we know what kind of data protection mechanisms and other security measures ought to be taken. Furthermore, the used standards and protocols should always concern security mechanisms that are pervasive, simple, scalable and easily managed (JFC2). Knowing this we should bear in mind which standard can be used and which one cannot.

A note in the context of using Jericho Commandments in this area: It is inevitable to also include JFC7. By improving the security with using appropriate standards and protocols, the mutual trust level improves. It is important a system should be secure by default or “out of the box” by using inherently secure protocols only that are open, free, interoperable and standardised. (Forum 2006e; Forum 2008d)

We observe works by other members of the Security and Innovation Research Centre at Capgemini. Without going into too many details such as encipher standards, a summarising study will be given with a focus on the available protocols and their (dis-)advantages.
Of course, in the view of the Jericho Context, especially if it comes down to end-to-end encryption, encipher standards are essential. Yet, it is out of the scope of this thesis to include all those details. See (Stan 2008a) for more details.

First, we look at different types of security: what kind of services and mechanisms are necessary to survive in a hostile world? From there, some relevant recommendations by other researchers will be taken into account, providing a short summary of their recommendations considering which service, mechanism and protocol ought to be used when and where. Starting with an overview from bottom (network layer) to top (application based security protocols), the list should not be considered a complete overview of all recommended security protocols. It should however give a sufficient envisioning of elements that arise when using a Collaboration Oriented Architecture. At the end, we have a quick look why the use of secure protocols and standards is so important.

Necessary services and Mechanisms for communicating over the Internet
Alina Stan (Stan 2008a) showed in her study that there are multiple mechanisms and services of security necessary to ensure secure communications over the internet:



Service	Mechanisms				
	Encipherment	Digital Signature	Access Control	Data Integrity	Authentication Exchange
Entity Authentication	Y	Y			Y
Data origin Authentication	Y	Y			
Access control			Y		
Confidentiality	Y				
Traffic flow Confidentiality	Y				
Data integrity	Y	Y		Y	
Non-repudiation		Y		Y	
Availability				Y	Y

Table 3.1: The relationship between security services and mechanisms (adapted after Stallings, 2005)

Table 2: Mapping Security Services and – Mechanisms. (Stan 2008a)

- **Services**: There is a set of services or security features recommended for use in order to be capable of communicating securely over the internet:
  - *Entity Authentication* or *Peer Authentication*: The ability to verify the identities of all entities involved in a message transmission. It ensures that the participating entities in a communication process are who they claim to be. This service is intended to offer protection against attackers who can impersonate authenticated entities and perform either a masquerade or an unauthorised replay of a previous connection.
  - *Data Origin Authentication*: The verification that the source of data received is the right one. This service must ensure that the connection is not interfered so a third party can masquerade as one of the legitimate acting parties for the purposes of unauthorised transmission or reception.
  - *Access control*: The ability to limit and control the access to host systems and applications via communication links. Entities that want access must often be first identified or authenticated.
  - *Confidentiality*: The protection of transferred data against attacks conducted by unauthorised entities. Transferred data should remain private and read only by the intended recipients. Therefore, the communications should be kept private from all parties except the ones entitled to receive them. Basically, confidentiality prevents unauthorised disclosure of sensitive information. This is done in several ways. (See (Metsaars 2008a; Stan 2008a) for more details.)
  - *Traffic flow confidentiality*: The ability to render the traffic flow invisible to the outside attacker.
  - *(Data) Integrity*: Integrity assures that transferred messages are received as they are sent, with no duplication, insertion, modification, reordering, or replays. In addition, deletion or destruction of data is included in this service, so all the transferred data should arrive to the receiver. Thus, this service prevents the unauthorised alteration or destruction of transmitted data by unwanted entities.
  - *Non-repudiation*: Non-repudiation service refers to the prevention of denial by an entity (the sender or recipient of a message) that has taken a particular action, such as sending or receiving a message. With this service, the receiver can prove that the alleged sender in fact sent the message and vice versa.
  - *Availability*: Availability characterizes a system with resources that are always ready to be used. In the context of communications over Internet, this means that whenever information needs to be transmitted, the communication channel is available and the receiver can cope with the incoming data. This property makes sure attacks cannot prevent resources from being used for their intended purpose.
- **Mechanisms**: The services will need certain tools or basic mechanisms to be functional, such as:
  - *Encipherment*: Encipherment or the art of cryptography is one of the most significant mechanisms, which is defined by Schneier (1996) as the science of keeping messages secure. See also (Stan 2008a) for more details about different cryptographic algorithms such as public/private key, shared key, hashing, et cetera.
  - *Digital Signature*: The usage of a digital signature in a document provides the capability to check whether the sender of the message is the real sender or a bogus entity. It also gives the capability to check whether the document has been tampered with. See (Stan 2008a) for more details.
  - *Access Control*: It limits and regulates the access to critical resources. This is done by identifying or authenticating the party that requests a resource and checking its permissions against the rights specified for the demanded object. See (Stan 2008a) for more details.
  - *Data Integrity*: The assurance that transferred messages and other data are received as they are sent, with no duplication, insertion, modification, reordering, or replays.
  - *Authentication Exchange*: The exchange of authentication credentials in order to check whether the person is who he says he is.

(Metsaars 2008a; Stan 2008a)
Besides this list of services and mechanisms, A. Stan mentioned three classes of security mechanisms:

- **Mechanisms for attack prevention**: The mechanisms in this class contain ways of preventing or defending against certain attacks before they can actually reach and affect the target.
- **Mechanisms for attack avoidance**: The mechanisms in this class assume that an intruder may access the desired resources but the information is modified in such a way that makes it unusable and invaluable to the attacker.
- **Mechanisms for attack detection**: The mechanisms in this class contain ways of detecting an intruder that has or is about to violate a security policy.

(Stan 2008a)

Which protocol to use where? – Overview of different layers
In this chapter we see which protocol can be used, and where. Again, this will be a summarising study of other people's work, in order to present a brief overview.[64]
In this study, we do focus on collaboration via the Internet and thus by use of the TCP/IP stack. As one can see in Figure 16, there are different levels where the security could be applied:

The IPsec protocol is transparent to end users and applications, and provides a general-purpose solution. Further, IPsec includes a filtering capability so that only selected traffic need incur the overhead of IPsec processing. (Stan 2008a)

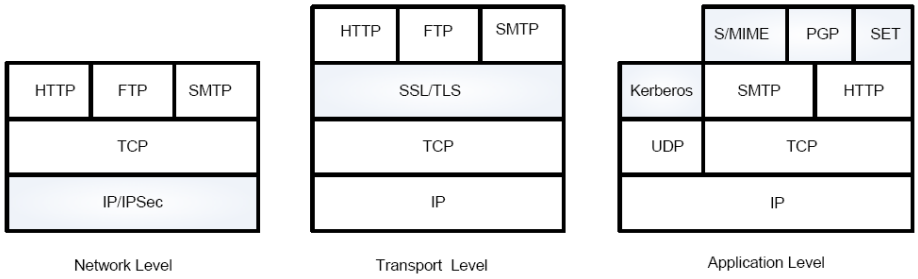


Figure 16: The different levels of protection mechanisms. (Stan 2008a)

Another option to provide a general-purpose security solution is to implement security above TCP (above the Transport Layer) by using, for instance, SSL or TLS protocol. Stallings (2005) specified there are two implementation choices for this protocol: SSL/TLS can either be part of the underlying protocol for providing full generality or it can be embedded in specific packages (e.g. Netscape and Microsoft Explorer browsers have options for this protocol, as well the Web servers). (Stan 2008a)
In the third picture in Figure 16, application-specific security services are illustrated, that are embedded within particular applications. These services can be tailored to the specific needs of any given application. A few protocols will be shortly noticed and described considering XML and other text-based security protocols. These are also application level based. See section 0 for some important backgrounds on SOA and security. (Stan 2008a)

Network level security: overview and recommendation of IPsec
When one uses network security level protocols, there will be a set of advantages over using transport or application level security protocols:

- **Transparent:** The traffic is completely transparent to the sending and receiving entities at the application level. One does not have to alter the application thanks to the transparency.
- **Encrypted:** The data remains encrypted in transit, along with the TCP header (and the IP header if used in tunnelling mode) so it will be secure when on its way.

[64] For more information on the TCP/IP stack, see Computernetworks Fourth Edition from Andrew S. Tanenbaum. Or look at http://en.wikipedia.org/wiki/TCP/IP_model for a more easy approach.

- **Any kind of traffic:** IPsec can secure almost any kind of network communication.

(Stan 2008a)

There are also a few problems with network protection: It will be hard for a firewall to see what kind of packets are coming through, depending on the used protocol. Furthermore, it is very hard to audit the security and traffic on the network for the same reason. (Stan 2008a) The usage of this protocol asks tremendous processing power on high-speed network connections. (Oppliger 1998)

One of the most complex and known protocols is IP Security (IPsec). It is actually a suite of protocols that is used for securing the IP fragments. Without going into the details of the suite: it is capable of delivering some significant security features such as:

- **Data Source Authentication**: Ensures that the communication takes place with a client that is authenticated and authorised for communication.
- **Access control**: Ensures that the communication occurs with a client that is IPsec-enabled.
- **Integrity**: Ensures the received data packets are identical with the data packets sent by the data source. Also assures the user that these packets have not been altered.
- **Anti-replay protection**: Verifies that no redundant data packets are received.
- **Confidentiality**: Enables encryption of transmitted data so, that the data remains confidential in traffic and that protection is offered against eavesdropping, it also provides the possibility to encrypt the IP packet header.
- **Key management**: offers secure exchange of keys.

(Stan 2008a)

The question that should rise now is "Is this a protocol which we can use in a de-perimeterised environment?"
The answer is "Yes and no". Drs. A. Stan (Stan 2008a) recommended in her thesis, *'Secure Communications'*, that IPSec can be used in transport mode to secure the communications across internet between two hosts if the used application(set) does not support SSL/TLS. Yet, this is dependant of the amount of resources available to the organisation that likes to secure their communications. Because of the complexity of IPsec, she also recommended there should be a set of tools to monitor IPsec, to see whether it is implemented properly. That is vital because of IPsec's complexity. (Stan 2008a)
The Jericho Forum recommends IPsec usage in (Simons 2006) for a de-perimeterised environment, only in the following cases:

- **Site-to-site connections or island-to-island connections**, where an area of secure connectivity is connected to another area of secure connectivity. In the transition to a de-perimeterised architecture this will prove a useful tool.
- **System-to-system,** Here, a system that requires to be semi-permanently connected to another system could validly use a VPN tunnel.

It should never be used as an addition to secure protocols that are insecure in themselves! See for more details: (Oppliger 1998; Tanenbaum 2003; Stan 2008a).[65]

Transport level security: overview and recommendation of SSL/TLS
Transport level security is about security level protocols at the transport level. This means that the destination and traffic information (TCP and IP headers) is visible to others, but the data itself is encrypted. It must be pointed out that the protocols on this layer are not transparent to

[65] For more details on IPsec and a good converging study and a set of recommendations of how to use IPsec in a Jericho environment, see Stan, A. (2008a). Jericho in depth, Secure Communications. Utrecht, Capgemini.

applications. One can replace all the TCP commands with SSL commands to use the protocols. (Chou 2002)
Two protocols, recommended by Stan, that secure the transport layer are the Secure Socket Layer and Transport Layer Security (SSL/TLS). The primary goal of these protocols is to provide privacy and data integrity for transferred data between entities. (Stan 2008a)
SSL 1.0 was created by Netscape and later enhanced by Microsoft (SSL 2.0). After that SSL 3.0 came out with a few efficiency upgrades and data compression support. TLS has been created by the Internet Engineering Task Force (IETF) based on the SSL 3.0 specifications (T. Dierks 1999; Blake-Wilson 2006; S. Santesson 2006; Santesson 2006; T. Dierks 2006).
Both protocols are actually stacks of protocols at the transport level (above TCP, see Figure 16). Both of them have the security features:

- **Position:** SSL/TLS works between the application and transport layers of the network protocol stack to ensure security of applications on the transport layer.
- **Characteristics:** SSL/TLS provide private, reliable, and non-forgeable conversations between two communicating processes.
- **Two-way:** Basically, SSL/TLS provides client-side and server-side authentication, confidentiality (encryption of the messages) and message integrity.
- **Multiple applications:** While frequently associated with web-based transactions, SSL is not limited to securing the HyperText Transfer Protocol (HTTP). Any upper-layer protocol or application that relies on TCP can employ the security services provided by SSL (e.g. news, e-mail, the File Transfer Protocol, Telnet, the Network News Transfer Protocol, the Internet Message Application Protocol, the Internet Relay Chat, and the Post Office Protocol 3).
- **Non-secure networks:** The communication can travel over non-secure networks by use of SSL or TLS.

(Stan 2008a)

Whenever a protocol like this uses the SSL protocol to secure it, then the S is appended in front of the protocol: SFTP, SHTTP, et cetera.
This protocol is recommended by Stan to be used as the end-to-end security protocol -if the application supports it- Thanks to multiple advantages such as partial encryption, anti phishing techniques by using certificates, the available compression, the openness of the standard, the maturity of the standards, the interoperability, easy configuration and the already named security features. See for more details (Stan 2008a).

Intermezzo 6, part one: Security below the network level: Wireless exposed.

There are two layers missing in the enumeration: The Physical Layer and the Data Link Layer from the TCP/IP point of view (Tanenbaum 2003).
These two are relevant if it comes down to regulating the traffic over a wire and optimize its bandwidth, but even more so if it comes down to the protection against eavesdropping when one tries to plug into the cable. That rarely happens nowadays, though.
In a wireless network it happens all the more: One can easily use a mobile device equipped with a Bluetooth adapter or a Wi-Fi system and scan for active PANs or Wireless LANs. If these networks are not protected by any protection scheme, like a passkey (passkey in Bluetooth, WEP, WPA (2) in Wi-Fi) and a MAC-access control list (WiFi routers and access points can often setup one) or 802.1x authentication, it becomes easy to listen in and even manipulate the messages by use of various attacks. (Tanenbaum 2003)
The Jericho Forum has released a position paper (Forum 2006g) on this topic and showed there are three different problems:

"1. Protection of the air-interface against unauthorized usage;
- In the public space the protection and generation of revenue
- In the corporate space, the protection against intrusion inside the corporate boundary
2. Authorization of the user to make a connection into the corporate WAN;
3. Privacy and confidentiality of data transferred over the connection."

(Forum 2006g)

Intermezzo 6, part two: Security below the network level: Wireless exposed.

They also came up with a set of solutions:

"1. Companies should regard wireless security on the air-interface as a stop-gap measure until inherently secure protocols are widely available
2. The use of 802.1x integration to corporate authentication mechanisms should be the out-of the box default for all Wi-Fi infrastructure
3. Companies should adopt an "any-IP address, anytime, anywhere" (what Europeans refer to as a "Martini-model") approach to remote and wireless connectivity.
4. Provision of full roaming mobility solutions that allow seamless transition between connection providers"

(Forum 2006g)

See for the COA v2.0 version https://www.opengroup.org/jericho/publications.htm, no changes have been made in the v2.0 version compared to this one. One additional paper has been released on this subject, which is summarized in the COA V2.0 update of paragraph 3.9.

A note of importance; SSL/TLS can work with a single authentication (only server side) or with a double (both server and client need to authenticate themselves with a certificate). The latter needs to be forced. There is one problem: TLS 1.1 and SSL 3.0 are not interoperable: So one of the two standards must be chosen if Transport Layer security is opted for as the appropriate level to secure the communications. (Stan 2008a)
Now, one could ask himself "Which protocol is best to use? SSL/TLS or IPsec?" This depends on various reasons and it is wise to consider the study of Drs. Stan (Stan 2008a) on this field.
The Jericho Forum also recommended the usage of Secure Shell (SSH) in one of their position papers (Forum 2006e) as a protocol for establishing communications over an insecure network. See for more details: (Forum 2006e; J. Schlyter 2006; S. Lehtinen 2006; T. Ylonen 2006a; T. Ylonen 2006b; T. Ylonen 2006c; T. Ylonen 2006d).

Application level security
There are many standards and niche-protocols at the application level security that are important in certain cases (see section 0 for a few examples). However, it is outside the scope of this thesis to name them. There are three fields worth mentioning in application level security, even without noticing any protocols: Authentication and authorisation, message protection and message filtering:

Application level security – Authentication and authorisation
This is a field where more research needs to be conducted. Even though it is overshadowed by larger Identity Management systems, there is currently no study that recommends a protocol on the field of authentication and authorisation. Of course, there are some useable standards and mechanisms (Jason Hogg 2005), but none of them have been examined in a Jericho context. There are a few Identity Management and authorisation recommendations; they are reviewed in section 2.6.7. The standards that can be used to describe the identities and manage them are reviewed in that section as well.
A by the Jericho Forum considered recommendable security standard in the field of authorisation is XAML (eXtensible Access Control Mark-up Language) (Forum 2008e). See section 0 for more details.

Application level security – Message protection
More research also needs to be done on the field of message protection. At this moment, no standards apply in a Jericho environment. This means we cannot recommend or expect any specific protocol in a setting of a Collaboration Oriented Architecture. Yet there are some guidelines to what may help and what may not. In (Jason Hogg 2005), a decision matrix is described for choosing the adequate security mechanisms to assure message protection for

transmitted data over the Internet. Drs. Alina Stan modified this message protection decision matrix in her work (Stan 2008a) and came up with a new overview. It is demonstrated in Table 3.
A currently available message protection standard is web-service security. More is to be found in section 0.
Drs. Stan also gave a further outline of message protection measures in the form of encoding, digital signatures, et cetera. See (Stan 2008a) for more details.

Application level security – Message filtering
The last important application level security mechanism to name is Message Filtering. The filtering discussed here is different from the packet filtering measures: they can only see little information such as the source's IP and port, the destination's IP and port, errors and used protocol. If they are stateful, they also scan for sequence numbers and flags. At best, it is visible what kind of packet is being used (TCP or UDP) and the filter reads the flags and sequence numbers - if they are stateful and Transport level-based. Message filtering is on a much higher level. Instead of network or transport level, it is at the application level. This means the content itself is inspected by measures such as:

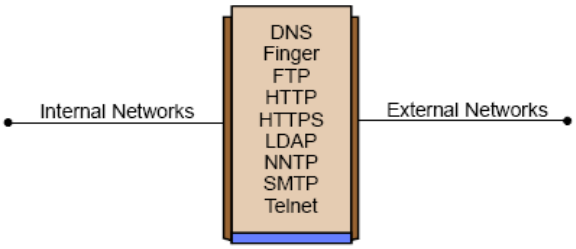


Figure 17: Typical Proxy-agents. (John Wack 2002)

- **Application-Proxy Gateway Firewalls:** All network packets are routed by the firewall and each of them is inspected for the protocols used. Each of the allowed protocols has its own proxy agent connected to the firewall. The proxy agent determines, based on the content of the message, a set of configurable rules and, if necessary, any means of authentication, whether the content is allowed to pass or not. This solutions has two objections: It is a very cpu-intensive solution, because it has to check each single packet. Moreover, there must be a software agent supplied to each protocol, while vendors tend to miss some of them. (Nelly Delessy-Gassant ; John Wack 2002)
- **Web service related filter mechanisms such as XML Schema, WSDL, Xpath et cetera:** See section 0 for more details.
- **Port Knocking:** An interesting addition to the firewalls is Port Knocking. It refers to a method of communication between two nodes in which information is encoded, possibly encrypted, into a sequence of port numbers. This sequence is named the knock. The server presents no open ports to the public and a port knocking daemon is monitoring all connection attempts. The client initiates connection to the server by sending SYN packets to the ports specified in the knock. Whenever the client gives the right port sequence, the daemon decodes a valid knock triggering a server-side process. (Maddock 2004)
- **(Dedicated) Proxy servers:** These applications retain proxy control, but without the firewall functionality. They can filter any kind of content based on the applications and protocols for which they are built. They do this the same way as the described proxy agents. (John Wack 2002)
- **Intelligent Application Gateways:** This is a term used by Microsoft for one of their products that is more than just a gateway: it is equipped with Application Optimizers that filter and use compression to allow more efficient communication and it provides a SSL VPN, a Web application firewall and end-point security management that enable access control, authorization and content inspection for a wide variety of line-of-business applications.[66]

[66] http://www.microsoft.com/iag, visited on the 1-07-08

Gene Spafford on secure systems

"As Gene Spafford, Director, Computer Operations, Audit, and Security Technology at Purdue University put it: "The only system which is truly secure is one which is switched off and unplugged, locked in a titanium lined safe, buried in a concrete bunker, and is surrounded by nerve gas and very highly paid armed guards. Even then, I wouldn't stake my life on it…" (Stanton 2005)

However, the Jericho Forum does not want to rely on these mechanisms alone. The focus will be at information protection and trust management. See also section 2.6.3.

Security Consideration	Data Integrity	Data Origin Authentication	Data Confidentiality
Verification if the contents of a message were not altered in transit.	Allows verification that a message has not changed in transit.	Supports the ability to verify that a message has not changed in transit and verify the origin of a message.	Encryption does not prevent the contents of a message from being altered.
Verification of the data source authentication and integrity of the message (has not been altered in traffic)	Allows verification that a message has not been changed, but this does not necessarily imply that the receiver can verify the source of the data.	Supports the ability to verify that a message has not changed in transit and verify the origin of a message.	Encryption does not prevent the contents of a message from being altered.
Restriction of the access to the contents of a message to authorized users only.	Does not provide the ability to protect message contents from unauthorized users.	Does not provide the ability to protect message contents from unauthorized users.	Confidentiality can be used to encrypt the contents of a message so that only authorized users can view the message contents.
Authentication is implemented based on shared secret between the entities participating in the communication. Prevention is required against attackers who want to recover the shared secret	Generating signatures based on shared secrets that may have low entropy leaves the message vulnerable to offline cryptographic guessing attacks; instead, direct authentications mechanism can be used	Generating signatures based on shared secrets that may have low entropy leaves the message vulnerable to offline cryptographic guessing attacks; instead, direct authentications mechanism can be used	Encryption combined with data integrity and data origin authentication can be used to protect the shared secret.
Implementation of message replay protection for preventing an attacker from maliciously replaying messages. . Replay detection depends on the ability to uniquely identify messages.	This option is often implemented using a hashing function that provides a unique identifier that can be used to determine if the same message is received multiple times.	This option is often implemented using a hashing function or digital signature that provides a unique identifier that can be used to determine if the same message is received multiple times.	Not applicable.

Table 3: Overview of message protection. (Stan 2008a)

When is a system (fully) secure?

So the two questions that arise while reading, are: "when can a system be considered secure?" and "can a system be fully secure?". Before we can answer any of those questions, we first have to understand what system is mentioned. Such a system can be defined as follows:

> *"A system can be seen as an information system that is a subsystem to an organisation with the goal to support that organisation with:*
> - *Saving new facts for a longer period of time (days, weeks, months)*
> - *Maintaining those facts*
> - *Optionally: start actions based on those facts*

- *Giving mechanisms to retrieve the facts*

The system can consist of multiple entities (or just one), which can be centralised or distributed."[67]

Starting with the first: One can and will always ask oneself when such a system is fully secure. The answer is rather tragic: never. One can always compromise data. The only problem is that we cannot fully prove this. Bruce Schneier, a famous security specialist, said in (Schneier 2003) that it would not be possible to evaluate the effectiveness of a security countermeasure. In fact, that it would not be possible to see if the system would always be secure. That is because we can only check on the systems security by attacking it. If the attack fails, the system is secure against only that specific attack for sure. It does not say if it is secure against an improved version of that attack or any other type.

One could argue, by means of the published criteria by the United States Department of Defense (US DoD) in their report (al 1985) "Trusted Computing Evaluation Criteria", later named "the Orange Book", that there is no system fully secured[68]. This can be done by the following line of reasoning:

- In order to be fully secure, a system needs to meet the A1 status, as defined in (al 1985).
- This means that every entity (See our definition and the criteria in (al 1985)) needs to meet the A1 status.
- There are only three components in the world that reached that status and all of them are network processors.
- We cannot create a digital information system, as in our definition, that will save new information for months based on network processors.
- There are no systems by our definitions that are secure.

Intermezzo 7-part one: Surf the web safely and secure in a de-perimeterised world: Internet Filtering and Reporting

The Jericho Forum released a position paper (Forum 2006d), in which they made a very important point. While surfing the net, there are three problems that exist, even in a de-perimeterised world:

"1. Ensuring that where you browse is in line with the stated (corporate, country, personal or home) policy on web browsing
2. Ensuring that what a web server delivers back is free from malicious content
3. Ensuring that all end-devices, no matter where, or how they are connected are protected"

In that same position paper, they also came up with the answer to these problems. Until the de-perimeterisation is completed, one will have to use either an internal internet Filtering and reporting service or from a third party. When completely de-perimeterised it will be a distributed service. The Internet Filtering and Reporting service should be capable of the following:

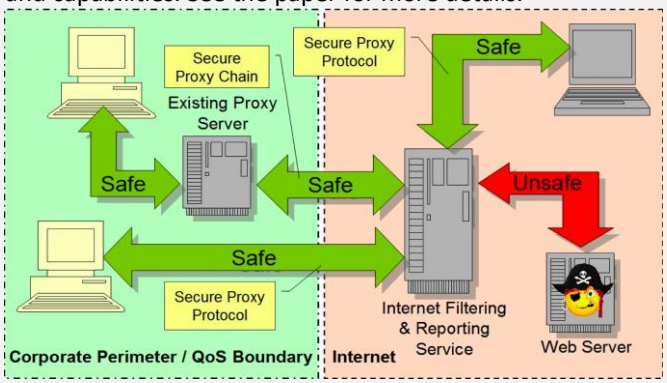
- Using the credentials to use credential based rules,
- define access by website category and time of day, total access time, etc.,
- the ability to force redirection to an Acceptable Usage Policy,
- the ability to show appropriate denial screens,
- database-based url-filtering, with usage of wildcards,
- sufficiently granular categorisation of sites,
- proxy identification mechanisms,
- HTTPS scans for malicious content,
- blocking/filtering by attachment,
- specific content blocking on a site,
- the ability to white- and blacklist sites.

[67] Definition is loosely based on the "Fully Communication oriented information Modelling by Bakema.

[68] See also intermezzo 8 for more details. Also a little note beforehand: in that same intermezzo we will see that there are some criteria on the Common Criteria.

Intermezzo 7-part two: Surf the web safely and secure in a de-perimeterised world: Internet Filtering and Reporting (REV2.0 ADDITIONALS)

In their newer release of that same paper[1], they also included a model of the filtering architecture (which is shown below) and a set of details considering the filtering features and capabilities. See the paper for more details.



[1]Ne releases can be found at: https://www.opengroup.org/jericho/publications.htm

Knowing there cannot be a (multipurpose business information) system that is fully secure, does not mean we have to worry all the time. Remember the commandments named in the beginning of this section? JFC1 automatically says that no one needs a completely secure system for most of the time. This answers the second question: A system is 'fully' secure when secure enough for the asset at risk. Yet, it is important to never forget about the 'context at your peril' when talking about the security of systems (JFC2).
When one has to protect crucial assets, he would like his system as safe as possible. Even though one cannot make it fully secure, each added measure helps, so does Bruce Schneijer say in (Schneier 2003).

With this, one could set up a list of criteria a system, based on the Jericho principles, should comply to in order to be secure enough for any asset at risk. Based upon the classification system used to classify the asset at risk to a certain risk level, one could formulate levels of security: The traffic light protocol used at Eli Lilly, which is based on the G8 traffic light protocol (Seccombe 2007) is an example. Within those levels of security, one could specify the minimum security measures that have to be taken for the asset at risk.
So to answer both of the questions:

- A system can never be fully secured.
- A system can be secure if the right security measures are taken considering the asset at risk, considering the context at ones peril.

Summarising and further Interest
A lot of very interesting topics have been quickly touched in this section. The following issues have caught the attention:

- The Jericho commandments that are important for the usage of protocols.
- The mechanisms one can use to secure communications.
- The Network level (IPsec) and Transport level (SSL/TLS) security protocols that are evidently important.
- The interesting fields of application layer security.

Even though it is outside the scope of this thesis to describe the details, one could look at the literature list and at the work of my peers (Paul Metsaars and Alina Stan) to understand more of the security threats and measures one could take.

COA V2.0 UPDATES:

The following relevant additions have been made in the COA 2.0 revision:

- **The issues and recommendations around the usage of VoIP in a de-perimeterised world**, in: "Position Paper VoIP in a de-perimeterised world", which can be summarized as follows:
    - *VoIP is insecure and used a lot.* VoIP is used a lot and will be used even more due to cost savings. However, it is very insecure: interception, recording/replay etc. can happen anywhere on the network. Many of its implementations do not have security built-in.
    - *The usage of proprietary protocols:* VoIP often uses proprietary protocols to secure inter-organisational communications which still do not provide a fully secure VoIP.
    - *Recommendations.* The Jericho Forum emphasizes the need for open standards and the need for "secure out of the box" implementations, in which all components in a VoIP implementation will be secure according to an industry agreed profile. The devices shoulde be capable of receiving security updates, allowing for mutual user authentication and surviving a connection to the raw internet by usage of the inherently secure protocols.

  See for more updates on this subject: section 3.9.2.
- **Introduction of the client to service VPN** in the new revision of "(The Need for) Inherently Secure Communications": As an addition to the statements above, the Jericho Forum advertises for a client to service VPN, in which applications use built in tunnel capability so that each protocol is isolated and only services/prots in use are exposed. See the paper for more details.

The publications can be found at: https://www.opengroup.org/jericho/publications.htm

2.6.5. Policy management

Introduction

As JFC5 says: "All devices must be capable of maintaining their security policy on an un-trusted network." An important topic to discuss should be policy management. The first question then is "What kind of policies should be managed in a Collaboration Oriented Architecture?" The second question should be "How can one manage these policies? ".

The policies inside a Collaboration Oriented Architecture

The answer is partially found in (Forum 2007c): This paper argues that the Jericho Forum philosophy leads to the use of information access policies. Every object that needs protection will need an information access policy.

Knowing there are many different objects, it would be out of the scope of the thesis to exactly specify the necessary policies. It is doubtful whether such an enumeration could even be possible. There are plenty of different objects for just a single corporation to have policies on. Moreover, what to think of the policies that will have to be enforced in a collaboration? There are lots of different collaborative relations. It is however obvious that two competitive parties have a completely different relationship than two complementary parties. Beside the different roles, one should also consider different branches. For instance, the automotive industry has a completely different set of policies than public healthcare.

Still, one could try to answer the question from a broader perspective. The answer is then also found in the ISO 27001 (and ISO 27002 or ISO17799) standard(s). The standard is created by

ISO (the International Organisation for Standardization) and IEC (the International Electrotechnical Commission) and focuses on information security.[69]
There are of course more fields where a collaborative party would like to enforce policies, besides information access policies, yet those are outside the scope of this thesis.[70]

Intermezzo 8: Evaluating security: first "the orange book" later the Common Criteria

The orange book or "Trusted Computing Evaluation Criteria" was designed by the US DoD in order to evaluate computer security products proposed for US government use. It supported the following evaluation classes:

- D – Minimal Protection
- C1 – Discretionary Protection: Identification and authentication and DAC. Minimal Assurance.
- C2 – Control access protection: Adds object reuse and auditing, More testing requirements, Windows NT 3.5 evaluated C2
- B1 – Labelled Security Protection: Adds MAC for some objects, Stronger testing requirements. Information model of security policy. Trusted Unixes tended to be B1
- B2 – Structured protection: MAC for all objects. Additional logging. Trusted Path. Least privilege. Covert channel analysis, configuration management, more documentation, formal model of security policy
- B3 – Security Domains: Implements full RVM. Requirements on code modularity, layering, simplicity. More stringent testing and documentation.
- A1 – Verified protection Same functional requirements as B3 Significant use of formal methods in assurance. Only three products made it to this standard.

(al 1985; Borisov 2007)

The process of evaluating the products by the NSA according to the Orange book took way too long (3 years or more). Other countries had their own validation criteria. (Anderson 2008)
Later on, the ITSEC proposed to let the vendor pay for the evaluation instead of the government, making it cheaper (and less secure: the vendor would seek the easiest way to get certified). In 1994-1995 the governmental instances got rid of their own evaluation criteria and worked together on the Common Criteria to be the new evaluation criteria.[1] (Anderson 2008)
In (Anderson 2008) Anderson came up with a few important issues around the CC and the orange book such as: They do not take human and organisational elements for and in a system. Furthermore, the CC skips a lot of important issues as crypto algorithms and evaluation methodology etc.
Yet, the CC, revised as ISO 15408, would seem to be promising according to the Jericho Forum, since the new CC allows one to create a flexible technology independent standard. See also section 2.6.10 for more details.(David Lacey 2006)

[69] See Intermezzo 9 for more details on the standard.

[70] Yet, there are some measures and frameworks named in section 2.4.2 such as COSO, which gives policies and control objectives as well.

ISO 27001 and the COA framework

In (Forum 2008e), the Jericho Forum named the IS 27001 as a valuable element. That is why the core of policy management in this thesis is based on that standard.

Intermezzo 9: Information Access Policies and ISO 27001/ 27002

In order to manage Information Access Policies, one should know what they are. The Jericho Forum uses the following definition in (Forum 2007c):

> *"A security policy is a rule that an organisation must follow in order to meet its security objectives. An Information Access Policy is a particular type of security policy specifically related to the security of information and its underlying data. There are many types of information access policy, some examples are given below:*
> - *'Personnel information shall be readable by the subject, the HR department, and the subject's manager'. This is a business-oriented human-readable policy. It is generic, in that it can be applied to many different assets.*
> - *A file ACL is a machine-readable policy. It is infrastructure-oriented and applies to just a single object.*
> - *'Users must not install applications on their laptops without permission from the security department'. This looks like the previous business-oriented policy, but note that it does not actually say what applications are permitted and which are not, nor does it even give criteria for permitting an application. Rather, it describes a procedure for obtaining permission ('ask the security department').*
> - *A software or DRM license is a form of policy. This is a machine-readable business oriented policy and may specify controls over copying and expiry."*
>
> (Forum 2007c)

The Jericho Forum sees a policy statement as critically important in controlling organisations and computer systems. There are two types of policy statements: human readable and computer readable. Furthermore, policies vary in scope: from generic policies applicable to multiple objects, to specific policies applicable to a single object.
More about policies is found in section 0.
So how does this relate to ISO 27001/27002? Both standards have been quickly covered in section 2.4.2. They both handle the issues around information security, which is achieved by implementing a suitable set of controls, **including policies**. Those information access control policies that are described in or can be inferred from these standards are the interesting ones for the thesis.

Getting back to the policies on the field of information security: 'Easy2Solve' created an ISO 27002 TOOLKIT[71] based on the mentioned standards. The toolkit categorises the different types of policies that are necessary in chapters and sections. Each category could be viewed from a Jericho perspective by the JFCs, and the position papers, focussing around (Forum 2007c):[72]

- Classifying information and data:
  - *Content:* These policies are about defining information, labelling, storing and handling classified information, isolating and classifying Top Secret information and accepting Ownership for Classified Information and Managing Network Security.
  - The Jericho Perspective:
    - The JFCs: These policies are elemental: they immediately support JFC1. In fact, these policies are so important that the Jericho Forum enforces these policies by using multiple processes and different (software) systems that are described in section 2.6.6. These policies must be maintained by any device on any network (JFC5), which means that it should also be possible to create a mobile data classification system that uses

[71] Which can be found at: http://www.27000-toolkit.com/, visited at 02-07-08. One can use and see that toolkit to see what the exact details are of the policies.

[72] It is very important to understand that one does not need all of the named policies. The standard itself and the toolkit see all of the policies as optional for any enterprise. One can see this standard as a policy container, from which they can choose their own information access policies.

inherently secure protocols for a hostile environment. Also, this policy set links to JFC11. The issues around accepting ownership might help one to classify the data, yet it could give some trouble considering JFC8.
    - Position Papers: If we look at the policies from the perspective of the position papers, one very important policy can be identified: 'The acceptance of ownership of the information'. This allows one to be the owner and applies to a certain type of governance pattern for policies such as automated or workflow-based controls.
- Controlling access to information and systems:
    - *Content:* These policies are about managing access control (standards), unattended workstations, passwords, higher risk system access and things like controlling and restricting access to operating system software, documents and files.
    - The Jericho Perspective:
        - The JFCs: Most of these policies are in line with JFC3: 'Managing Access Control Standards', 'Managing User Access', 'Securing Unattended Workstations', 'Managing Passwords', 'Securing Against Unauthorised Physical Access', 'Restricting Access', 'Restricting Access' and 'Monitoring System Access and Use'. All the mentioned policies are about either securing the environment, or about informing users about the insecurity of the environment. 'Giving Access to Files and Documents' is directly in line with JFC11 and JFC10: the policy describes that access to information should be carefully controlled. 'Managing Higher Risk System Access' is in line with JFC1: the access controls need to be set in accordance with the value of the information asset. Yet, 'Managing User Access' could be in violation with JFC8 since the authorisation could be inside the locus of control of the users, while the policy controlling remote user access is indeed in compliance with this JFC.
        - Position Paper: Looking from the perspective of the position papers, we can see that most of the named policies here can use automated control, yet some of them such as higher risk system access, are more workflow-based in its control and will use time-limited permissions as well.
- Processing information and documents:
    - *Content:* This chapter of policies consists of nine different sections:
        - Networks: The policies around networks are about configuring, managing the network, accessing it remotely and defending it.
        - System operations and administration: These policies handle different procedures around system operations and administration such as 'Appointing System Administrators', the procedures around system administration itself and the management of electric keys.
        - E-mail and the Worldwide Web: The policies around this topic are about different procedures on the internet such as handling email, access to intra-, extra- and internet, usage of internet and its search engines, websites.
        - Telephones and Fax: These policies are about the usage of the telephone, videoconferencing facilities and the fax. There are extra policies on the field of misdirected information by fax and unsolicited faxes.
        - Data management: Policies about different aspects of data management such as setting up, managing and using data storage, databases, spreadsheets, drafts, version control systems, files and/or documents. There are also policies about linking information between documents and files, sharing data (on project management systems) and using data from third parties.
        - Backup, recovery and archiving: Policies on exactly the topics that are named in the title, including policies about restarting systems (as part of a recovery or plainly a restart).
        - Document handling: There are different policies around document handling such as 'Handling and Managing (printed) documents', 'non-repudiation methods', filing and transporting them, style of a document and 'what to do when receiving unsolicited mail'.

- Securing data: There are different policies in this field, using encryption and passwords, sharing and sending information (with or to third parties), handling and maintaining confidentiality from customers, deleting data created/owned by others and the printing of classified documents are simple examples.
- Other information handling and processing: this section handles the remaining policies on information and document handling, ranging from using dual input control to speaking to customers and media, from clear desk policies to travelling on business and checking customer credit limits. An important policy to this thesis is 'Misaddressing Communications to Third Parties'.

- The Jericho Perspective:
  - The JFCs, divided into sections:
  - The Network section is in line with JFC3, since it mostly considers the networks and thus the context of the information (-exchange). Some of the policies ('Managing Networks' and 'Configuring the Network') are also in line with JFC2; bordering the ease of managing it as well as scalability. If we look at JFC5, we can see that most of these policies will run on any mobile device, except for 'Accessing a Network Remotely': one might get into trouble if, for instance, a mobile device is unable to handle the protocols (in violation with JFC4) used on the network.
  - The system operations and administration section are mostly in line with JFC3. Some also linked to other JFCs such as the policy 'Controlling Data Distribution', which is more in line with JFC10 and JFC11. A policy like 'Managing Electronic Keys' is in line with JFC 2 (the PKI needs to be easy to manage), JFC3 (knowing the environment can do wrong with the private key) and JFC 6, 9 and 11 (the dual control). 'Managing System Operations' and 'System Administration' could both comply to, as well as be in violation of, JFC3 because the term "efficient" is added to the style of management of the security measures. 'Managing System Documentation', 'Scheduling Systems Operations', 'Scheduling Changes to Routine Systems Operations', 'Synchronising System Clocks', 'Commissioning Facilities Management', 'Responding to System Faults' and 'Monitoring Error Logs' are in line with both JFC3 and 2. 'Monitoring Operational Audit Logs' and 'Managing or Using Transaction/Processing Reports' comply to JFC2 and 3 and could comply to JFC6 depending on the interpretation and the support for semi public third party auditing. If we want to be capable of handling all of these policies on any device, we get into trouble considering the PKI management or the system operations: some devices may not be capable of handling this at all.
  - The E-mail and Worldwide web section is all about JFC3, dealing with context discovery or protection against the problems that can be caused by setting up or using intra-, extra-, internet and e-mail. If one wants to be able to use all policies on a mobile device, additional software is absolutely required.
  - The Telephones and Fax section: Most of these policies are in line with JFC3: they deal with preventing discovery of information security issues around the usage of telephone and fax. Some policies also hit other JFCs such as 'Recording Telephone Conversations', 'Giving Information When Ordering Goods by Telephone', 'Persons Requesting Information by Telephone' and 'Persons Giving Instructions over the Telephone': they could be in line with JFC6.
  - The section of data management consists mostly of policies that are in line with JFC9, 10 and 11. 'Updating Customer Information' considers JFC6 and 7. 'Managing Data Storage' and 'Setting up a new Folder/Directory' could however be both in line with, as well as in violation of, JFC5. It depends on their implementation. This is the same for policies 'Receiving information on Disks' and 'Setting up a new Folder/Directory', considering JFC2. Many of these policies might be hard to pertain on mobile devices. For instance: if one wants to create a new database on a laptop while stringent regulations are enforced around data protection, the laptop might not have the computing power to create, manage or access such a database.
  - The section of Backup, Recovery and Archiving consists mostly of policies in line with JFC 2, 3 and 11 considering their targets. They deal with the fact that it must be

manageable, with the dangers of the context and the safety of the files. Every device should be able to handle the policies (JFC5), based on the implementation.
    - The section of Document Handling consists of policies in line with JFC1: most of the document handling policies describe different security levels and a variable approach based on those levels. JFC3: most of the policies also consider the environmental hazard. JFC10: the policies always refer to the authorised personnel. JFC11: most of the policies also concern the safety of the documents and its security. The policies with only a little consideration of authorisation can be in line with or in violation of JFC8, depending on the implementation. Some policies such as 'Style and Representation of Reports' do not consider any JFC at all. Arguably, it raises trust and thus has a vague relation to JFC6. This may however be a little far-fetched.
    - The section Securing Data consists of policies, focussing on topics that consider JFC 9, 10 and 11. Many mobile devices could be in trouble depending on the implementation of the policies: using processor-intensive encipher techniques might be too much for a mobile device.
    - The section Other Information Handling and Processing: all of the policies concern issues around JF3. Most of them do not concern devices at all, or are designed only for specific devices, so there should be no problem applying JFC 5 in this section.
    - Position Paper: If we look at these sections from the perspective of the position papers, we see that different patterns of governance are a necessity: some policies can be handled with automated control while some pure analogue policies (such as 'Clean Desk' policy) clearly cannot be automated with the technology of today. In order to be capable of handling all these policies, we should have a system that expresses the rights and authorisations of a user in a more efficient way than the technology of today. Many of these policies have a high variety of user groups that are authorised to execute certain procedures. Another important issue on this field is the exchange of information. The papers describe that, whenever information is exchanged, the information access policies should be exchanged as well. This may become troublesome whenever one exchanges analogue data - printed materials.
- Purchasing and maintaining commercial software:
    - *Content:* There are three different sections of policies in this chapter:
        - Purchasing and installing software: The policies in this field are about specifying user requirements, selecting business and office software, using licensed software and implementing the acquired software (upgrades).
        - Software Maintenance and Upgrade: These policies consist of applying patches and upgrades to software, responding to vendor recommended upgrades, interfacing and supporting application software, support and the upgrades of the operating system, recording and reporting software faults.
        - Other Software Issues: The only policy here handles what needs to be done while the disposing of software.
    - The Jericho Perspective:
        - The JFCs, per section:
            - Most of the Purchasing and installing software policies are in line with JFC3, while some such as 'Using Licensed Software' and 'Implementing New Upgrade Software' also work with the principle of JFC2.
            - The policies around Software Maintenance and Upgrade are mostly compliant to JFC 1, 2 and 3. JFC1 is found in the policies around upgrades and patches: If the asset at risk or its supporting system requires an update, it should be updated or patched. Most of the policies show that security mechanisms should be pervasive. Furthermore, all of the policies warn or deal with assuming context at peril.
            - The Other Software Issues carry one policy around the disposal of software, which is not directly linked to any of the JFCs.

        - Position Paper: Looking at these sections from the position papers' point of view, we should understand it is important to make sure that the software can deal with the policy management and enforcement software when purchased, patched or updated.
- Securing hardware, peripherals and other equipment:
    - *Content:* There are seven sections in this chapter:
        - Purchasing and installing hardware: The policies in this section are about specifying information security requirements for new hardware and the detailed functional needs, installing new hardware and testing systems and equipment.
        - Cabling, UPS, printers and modems: There are different policies in this section such as 'Managing and Handling Power', 'Using Faxes, Modems, Printers' (networked and standalone) and 'Installing and Maintaining Network Cabling'.
        - Consumables: describing the control and handling of IT consumables such as removable storage media.
        - Working off premises or using outsourced processing: this section is about different policies such as 'Contracting or Using Outsource Processing', policies around laptops and portable computers, tele-working, moving hardware, usage of mobile phones and business centre facilities.
        - Using secure storage: policies around the use of lockable storage cupboards, filing cabinets, fire-protected storage cabinets and a safe.
        - Documenting hardware: this section contains policies around managing and using hardware documentation and maintaining a hardware inventory or register.
        - Other hardware issues: this chapter's closing section consists of different policies such as the 'Disposal of Obsolete Equipment', 'Insuring Hardware and Hardware Abroad', 'Clear Screen' policies, 'Logon and Logoff', answering machines, taking equipment off the premises, on- and offsite support, 'Cleaning of Hardware' and 'Damage to Equipment'.
    - The Jericho Perspective:
        - The JFCs per section:
            - The policies around Purchasing and Installing Hardware are mostly in line with JFC1 and JFC3: they aim at buying and installing the hardware with their purpose taken in mind and they warn for hazards and dangers or try to deal with these threats.
            - The section Cabling, UPS, Printers and Modems consists of policies that are mostly linked to JFC3: they warn, or take preventive measures, for certain dangers. Some of the policies are in line with JFC2, JFC10 and JFC11 if it comes down to faxing or using modems and DSL connections.
            - The section around the policies that consider consumables are mostly in line with JFC3: they deal with the usage and controlling of IT consumables such as removable storage media.
            - The section Working off premises or using outsource processing consists of policies that are mostly related to issues around the usage of mobile hardware. They are in line with JFC3 since they prevent or warn for hazards, errors, misuse or information security problems within that usage. Some of the policies such as 'Day to Day Use of Laptop/Portable Computers' are also in line with JFC11, since it is about protection of the data on the laptop.
            - The section Using Secure Storage are all in line with JFC1, 2 and 11: all of these policies take the asset that needs to be stored in account in order to provide certain scalability around the storage. Yet all data should be stored securely.
            - The section Documenting hardware policies is entirely in line with JFC3: it deals with context at peril.
            - The policies in the section 'Other hardware issues' are mostly in line with JFC3 for the same reason.
        - The Position Paper: most of these policies will be very complex to govern; in fact, it will be hardly plausible to ever automate all of them. Other means of management should be found for these policies.

- Combating cyber crime:
  - *Content:* This chapter consists of one section, with the policies: 'Defending Against Premeditated Cyber Crime Attacks', 'Minimising the Impact of Cyber Attacks', 'Collecting Evidence for Cyber Crime Prosecution', 'Defending Against Premeditated Internal Attacks and Opportunistic Cyber Crime Attacks', 'Safeguarding Against Malicious Denial of Service Attacks', 'Defending Against Hackers, Stealth- and Techno Vandalism', 'Handling Hoax Virus Warnings', 'Defending Against Virus Attacks', 'Responding to Virus Incidents' and 'Installing Virus Scanning Software'.
  - The Jericho Perspective:
    - The JFCs: All of the aforementioned policies are in line with JFC2: they describe how to make a pervasive and manageable set of security mechanisms against cyber crime. This automatically makes them in line with JFC3 as well, since they try to protect against dangers from the context at peril. Looking at JFC5, it is understandable that the devices must be capable of detecting the crime, withstanding it and following the policies around these subjects simultaneously. This might be quite an effort for a mobile device, which means one must look for a certain distributed form of security against cyber crimes on mobile devices.
    - Position Paper: Most of the detectable cyber crime can be fought with partly or completely automated policies, yet the language that will be used to express them must be extremely powerful.
- Controlling e-commerce information security:
  - *Content:* This chapter consists of one section with the policies: 'Structuring E-commerce Systems' including web sites and networks, 'Configuring E-commerce Web Sites' and 'Using External Service Providers for E-commerce'.
  - The Jericho Perspective:
    - The JFCs: The policies in this chapter are in line with JFC1; almost all of them show that the scope and level of protection can be variable depending on the asset at risk. These policies are in line with JFC11 as well, since they show the data must be secured.
    - Position Paper: It seems that the governance patterns for these policies should not be too hard to combine: most of them can be automated and some will have to follow a workflow-based control.
- Developing and maintaining in-house software:
  - *Content:* This chapter consists of the following sections:
    - Controlling Software code: the policies managing operational program libraries and program source libraries, and the controlling software code under development, program listings, source libraries and old versions of programs, are discussed.
    - Software development: Here, the policies 'Software Development', 'Making Emergency Amendments to Software', 'Establishing Ownership for System Enhancements', 'Justifying New System Development', 'Managing Change Control Procedures' and 'Separating Systems Development and Operations', are viewed.
    - Testing and training: This section consists of the policies 'Controlling Test Environments, 'Using Live Data for Testing', 'Testing Software Before Transferring to a Live Environment', 'Capacity Planning' and 'Testing of New Systems', 'Parallel Running' and 'Training in New Systems'.
    - Documentation: In this section, there is only a policy on documenting new and enhanced systems.
    - Other software development: a policy on acquiring vendor-developed software.
  - The Jericho Perspective:
    - The JFCs per section:
      - The policies in section Controlling Software Code touch JFC 6, since they try to create extra trust into the technology (software). Again, all of the policies are in line with JFC3 because they try to check for and guard against errors and hazards from the context at peril. They could be in violation of JFC8 depending on the implementation of the authorisation of personnel.

        - The policies in section Software Development: Most of the policies are in line with JFC10. That is, without data privacy and only for the insurance that the code will be correct. Depending on the implementation of the policies, they could be either in line with or in violation of JFC8.
        - The policies in section Testing and Training: Most of the policies are in line with JFC10, but without data privacy and only assuring that the code will be correct. Depending on the implementation of the policies, they could be either in line or in violation with JFC8. Some such as 'Capacity Planning' and 'Testing of New Systems', are in line with JFC3 since they warn or check against problems, hazards or threats from the context at peril.
        - The policy in section Documentation is in line with JFC3, since it warns and checks against problems, hazards or threats from the context at peril.
        - The policy in section Other Software Development cannot be linked to any JFC.
      - Position Paper: Most of the policies around software development could be expressed in a powerful language. Yet, some form of additional control and governance is necessary since these policies are going further than only a simple access policy. This automatically means that ACLs will not be powerful enough.
- Dealing with premises related considerations:
  - *Content:* This chapter consists of 3 sections:
    - Premises security: this section consists of the policies 'Preparing Premises to Site Computers', 'Secure Physical Protection of Computer Premises', 'Ensuring Suitable Environmental Conditions', 'Physical Access Control to Secure Areas' and 'Challenging Strangers on the Premises'.
    - Data stores: this section consists of the two policies 'Managing on-site Data Stores' and 'Managing Remote Data Stores'.
    - Other premises issues: The three policies here are: 'Electronic Eavesdropping', 'Cabling Security' and 'Disaster Recovery Plan'.
  - The Jericho Perspective:
    - The JFCs per section:
      - Premises Security: All policies in this section are in line with JFC3 since they either warn for or protect against threats, errors or problems in the context at peril. Furthermore, the policy around 'Physical Access Control to Secure Areas' could be either in line with or in violation of JFC8, depending on its implementation.
      - Data Stores: The policies in this section are in line with JFC1 (they consider the asset at risk and a certain differentiation in the level of protection), JFC3 (they warn for certain threats or problems coming from the context at peril) and JFC11 (considering data protection).
      - Other Premises Issues: All policies in this section are in line with JFC3 since they either warn for or protect against threats, errors or problems in the context at peril.
    - Position Paper: In order to maintain and govern these policies, different models of control are necessary. This asks for a more complex system. We however do not want it to be slow, especially when one needs to follow a disaster recovery plan. Besides, it will have to be capable of surviving these disasters in order to support, monitor and coordinate the systems and human resources when disaster strikes.
- Addressing personnel issues relating to security:
  - *Content:* This chapter consists of the following sections:
    - Contractual documentation: this section is about different policies such as: 'Preparing Terms and Conditions of Employment', 'Employing/Contracting New Staff', 'Contracting with External Suppliers/Other Service Providers', 'Using Non-disclosure-agreements', 'Misuse of Organisation Stationery', 'Lending Keys to Secure Areas to Others', 'Lending Money to Colleagues', 'Complying to Information Security Policy', 'Establishing Ownership of Intellectual Property Rights' and finally the 'Employees' Responsibility to Protect Confidentiality of Data'.

- Confidential personnel data: here are different policies such as 'Respecting Privacy in the Workplace', 'Handling Confidential Employee Information' and other issues around (personal) information exchange between employees.
- Personnel information security responsibilities: here are different policies such as the 'Proper use of Organisation Materials' (internet, computers, passwords, credit cards, telephone systems, information), 'Signing for the Delivery of Goods and Work Done by Third Parties', 'Approving and Authorising Expenditure', gossiping and disclosing information.
- HR Management: two policies matter here; 'Dealing with Disaffected Staff' and 'Taking Official Notes of Employee Meetings'.
- Staff leaving employment: this section consists of the following three policies: 'Handling Staff Resignations', 'Completing Procedures for Staff Leaving Employment' and 'Obligations of Staff Transferring to Competitors'.
- HR issues other: this closing section consists of a policy around recommending professional advisors.

- The Jericho Perspective:
  - The JFCs per section:
    - Contractual Documentation: all of the mentioned policies try to create a situation in which JFC6 and JFC7 can be applied, yet this section does not seem to work with the concept of trust.
    - Confidential Personnel Data: all policies try to create a situation in which JFC6 and JFC7 can be applied, yet this section does not seem to work with the concept of trust. These policies could be either in line with or in violation of JFC8, depending on their implementation.
    - Personnel Information Security Responsibilities: most of the policies here try creating a situation in which JFC6 and JFC7 can be applied, yet this section does not seem to work with the concept of trust either. Furthermore, some of the policies are in line with JFC1 since they use different levels of secrecy regarding the information as within 'Sharing Organisation Information with Other Employees'. Some policies such as 'Using Organisation Credit Cards' can be either in line with or in violation of JFC8, depending on the implementation of the authorisation.
    - HR Management: all of the named policies try to create a situation in which JFC6 and JFC7 can be applied, yet this section does not seem to work with the concept of trust. The policies speak of authorisation so, depending on how this authorisation is implemented, they could be either in line with or in violation of JFC8. The policy 'Taking Official Notes of Employee Meetings' is in line with JFC1, since it uses different levels of secrecy and different levels of appropriate protection.
    - Staff Leaving Employment: Both of the policies here try to create a situation in which JFC6 and JFC7 can be applied, yet this section does not seem to work with the concept of trust. Furthermore, they work with JFC10 in order to maintain data privacy.
    - HR Issues Other: the policies here do not directly link to any JFC.
  - Position Paper: Most of the policies in this chapter are about human resources and are quite complex to govern by means of a system. Again, a powerful language and system will be necessary.

- Delivering training and staff awareness:
  - *Content:* this chapter consists of two sections:
    - Awareness: The policies worth mentioning here are: 'Delivering Awareness Programmes to Permanent Staff', 'Third Party contractor: Awareness Programmes', 'Delivering Awareness Programmes to Temporary Staff', 'Drafting Top Management Security Communications to Staff' and 'Providing Regular Information Updates to Staff'.
    - Training: This section is about the policies: 'Information Security Training on New Systems', 'Information Security Officer: Training', 'User: Information Security Training', 'Technical Staff: Information Security Training' and 'Training New Recruits in Information Security'.

- The Jericho Perspective:
  - The JFCs: Both of the sections contain policies that focus on JFC3: they try to make the users aware or train the users to deal with information security hazards or threats from the context at peril.
  - Position Paper: The mentioned policies are about human resource, yet are easier to govern without automation (it will be harder with automation).

- Complying to legal and policy requirements:
  - *Content:* This chapter consists of the four sections:
    - Complying to legal obligations: this section deals with different issues such as complying to the data protection act or equivalent, general copyright legislation, database copyright legislation and copyright and software licensing legislation. It also has a policy around legal safeguards against computer misuse.
    - Complying to policies: This section consists of two policies: 'Managing Media Storage and Record Retention' and 'Complying to Information Security Policy'.
    - Avoiding litigation: this section holds the policies: 'Safeguarding Against Libel and Slander', 'Using Copyrighted Information from the Internet', 'Sending Copyrighted Information Electronically' and 'Using Text Directly from Reports, Books or Documents'.
    - Other legal issues: this closing section holds the policies: 'Recording Evidence of Incidents', 'Renewing Domain Name Licences – Web Sites', 'Insuring Risks' and 'Recording Telephone Conversations'.
  - The Jericho Perspective:
    - The JFCs: Most of the policies in these sections are to create a situation in which JFC6 and JFC7 could be applied, yet this section does not seem to work with the concept of trust. They work with compliancy or with setting up measures to be capable of being compliant. One could argue that, since many of the policies are about legislation, they are in line with JFC3 because they use the rules of the context at peril. Policies around the topics 'Complying to the Data Protection Act' or Equivalent' and 'Managing Media Storage and Record Retention' are in line with JFC9, 10 and 11, depending on the implementation.
    - Position Paper: Most of the policies could be governed by implementing business rules into the system. To make it workable however, one has to separate the administration points, decision points and enforcement points for these policies.

- Detecting and responding to IS incidents:
  - *Content:* This chapter consists of four sections:
    - Reporting information security incidents: this section contains policies about: 'Reporting of Information Security Incidents', 'Information Security Breaches', 'Reporting Information Security Incidents to Outside Authorities', 'Notifying Information Security Weaknesses', 'Witnessing an Information Security Breach' and 'Being Alert for Fraudulent Activities'.
    - Investigating information security incidents: this section consists of the four policies: 'Investigating the Cause and Impact of Information Security Incidents', 'Collecting Evidence of an Information Security Breach', 'Recording Information Security Breaches' and 'Responding to Information Security Incidents'.
    - Corrective activity: this section is about a policy named 'Establishing Remedies to Information Security Breaches'.
    - Other information security incident issues: In this closing section different policies are named such as 'Ensuring the Integrity of Information Security Investigations', 'Analysing Information Security Incidents Resulting from System Failures', 'Using Information Security Incident Check Lists', 'Establishing Dual Control', 'Detecting Espionage' and 'Monitoring Confidentiality of Information Security Incidents'.
  - The Jericho Perspective:
    - The JFCs per section:
      - Reporting Information Security Incidents: most of the policies in this section are aimed to create a situation in which JFC6 and JFC7 could be applied, yet this section does not

seem to work with the concept of trust. They also deal with environmental or organisational hazards or information security risks, which makes them in line with JFC3 and with JFC2, depending on the implementation.

- Investigating Information Security Incidents: the policies in this section deal with hazards and responses to information security threats and are therefore in line with JFC3.
- Corrective Activity: The policy in this section deals with hazards and responses to information security threats. It is therefore in line with JFC3. Furthermore, it could be in line with all of the other JFCs, depending on the implementation.
- Other Information Security Incident Issues: Most of the policies in this section deal with hazards and responses to information security threats and are therefore in line with JFC3. Policies 'Establishing Dual Control' / 'Segregation of Duties' are the direct enforcement of JFC10. Policies such as 'Monitoring Confidentiality of Information Security Incidents' can be in line with JFC 6, 7 and 8, depending on the implementation.
  - Position Paper: Since these policies are mostly about dealing with actual threats, a very powerful, fast and accurate system will be necessary, with the right measurements and expression language.

- Planning for business continuity:
  - *Content:* This chapter consists of one section with the following policies: 'Initiating, Assigning, Developing and Testing the BCP' (Business Continuity Project management), 'Training and Staff Awareness on BCP' and 'Maintaining and Updating the BCP'.
  - The Jericho Perspective:
    - The JFCs: The Business Continuity Plan (BCP) deals with issues around continuity when facing threats from the environment. This makes it in line with JFC3. Depending on the implementation of the BCP, it can be in line with all of the JFCs.
    - Position Paper: the policies around and/or coming from the BCP will be used in situations with high stress-levels and often heavily threatening incidents. This means that a very powerful, fast and accurate system is necessary, with the right measurements and humanly understandable expression language.

Looking back on all of the sections, we can see:

- **JFCs:** JFC 1, 2(partially) 3, 10 and 11 are covered by the information security policies based on the standards. The other JFCs can be partially or completely covered, depending on implementation.
- **Position paper point of view:** The most important recommendation from the position papers' point of view is the need for a powerful system and a good expression language.
- **Trust:** There is no real notion of trust in the ISO standards to such an extent that it covers JFC 6 and 7.
- **Authorisation:** There is no notion of authorisation, accounting, et cetera that the policies based on these standards will automatically cover JFC8.

How one can manage these policies:

The position paper (Forum 2007c) gives the following recommendations for policy management:

> *"-Information access policies must be expressible in powerful languages that can accurately capture the intention of the creator.*
> *-Secure systems need to separate out the administration points, decision points and enforcement points for information access policies.*
> *-Businesses need to adopt new techniques for understanding their security imperatives so they can be accurately encoded.*
> *-A set of interoperable global identifiers needs to be developed.*
> *-Where organisations exchange data, they should also expect to exchange information access policies covering how that data should be handled."….*

"-Organisations need to understand what their information access policies really are. It is important to realise that there will be many governance patterns for policies, here are some examples:

-Automated control. The owner of a data item specifies an information access policy about how it may be used. All holders of the item must consult the policy before they may give access to it.

-Workflow-based control. In this case it is not possible to specify a simple information access policy that a machine can follow, so the data owner (or his delegate) must be involved personally in the authorisation process.

-Accountability. In this case the data owner trusts a data holder to control access to his data, but retains the right to know who has accessed his data and why, and to hold accessors accountable for their access.

-Time-limited permissions. The data owner gives permission for a very short period of time after which the data holder must seek permission again.

-To allow such complex models to operate, systems must be able to separate information access policy administration (which will be done by the data owner); policy decision (which will be done by the owner or his delegate); and policy enforcement (which must be done by all data holders). This is the basis of standards such as XACML.

-Information access policies need to become more sophisticated than ACLs. Essentially they are specialised programs. Experiments have been conducted into expressing policies as proof obligations, for example. XACML[73] *is a valuable step in this direction, but only a step.*

-Means of managing identities between different attribute authorities. The Open Group's Common Core Identity standard is considering this issue."

(Forum 2007c)

The same paper states that efficient information access policy management is critical to securing an agile, rapidly changing enterprise. It argues that it will be quite hard to apply and enforce information access policies on mobile data if privacy and intellectual property rights issues are to be properly addressed.
With these recommendations and issues in mind, there is currently no answer besides the COA framework of how policies should be managed.

Getting back to the Jericho Forum Commandments
Before we jump to conclusions, we must look at the JFCs to see whether we can find or infer more relevant recommendations around policy management.

- **Variable scope and protection:** When we look at JFC1, it is to be understood that the policies should be variable in scope and protection. There is a lot of information that does not need the highest protection available. In fact, there are documents in which only a few details should be highly secured. This means that the policy management system should have a few crucial capabilities:
  - *Applying a single policy at many objects:* Many objects or "data containers" (such as files, documents, e-mails, database records) hold information which can be open and does not need to be confidential. This could allow us to use a single policy or a set of policies for more objects at once, since no real stringent control is necessary.
  - *Applying multiple policies at one object:* As we have open information, we also have confidential or top secret information (or any other "rank" in secrecy/confidentiality). This information should be strictly controlled by policies or sets of policies.
  - *Applying multiple/a single policy to a part of an object:* In fact, both of the situations described above may occur at the same time: a large document may contain much open information but simultaneously carry some confidential information. In such a case, it is a

[73] About XACML: see also section 0 for more details.

necessity that some parts of the information get additional (sets of) policies to control the information distribution/modification/creation/deletion, et cetera.

- **Simple and easy to manage:** JFC2 shows us that security mechanisms should be pervasive, simple, scalable and easy to manage. This means for a(n information security) policy management system that it should:
  - *Run on any hardware/OS that is used today:* In order to make it pervasive, the policy management system should have the capability to run on every hardware platform that is used by the modern de-perimeterised enterprise of today. How this is achieved, is left open by the commandments. This means that a distributed approach might do it (Since it will be quite hard to run a complete policy management system on a mobile phone with less processing power).
  - *be usable for sets of- or single objects:* In order to make it scalable, it should be capable of managing a single policy and multiple sets of policies.
  - *Be easy to manage*: It should be easy to manage, which means there should be an easy way of configuring, creating, modifying, deleting and managing the policies. How it ought to be done is left blank by the JFCs.
- **Tamper proof/ based on open inherently secure standards and protocols:** If we look at JFC 4 and 5, then we can see that the system should be based on open inherently secure protocols and a standard ought to be designed (or an existing standard may be used) that is inherently secure, open and usable on any device.
- **Auditable:** Looking at JFC6 and 7, we can say the system must have the capability of providing the assurance that the policy management system is correctly installed. Certain controls thus must be made possible within the system making it auditable.

Concluding:
As we look on policy management from the perspective of the ISO 27000 series and from the Jericho Forum's point of view, we find a large number of information access management policies to consider. These however do not cover the Jericho Forum Commandments fully. It is implementation specific whether all of the JFCs are covered or not. Furthermore, there is no notion of trust inside the ISO standards or inside a policy, that directly enforces it. The same goes for authorisation. We did however find the type of policy that should be managed: Information access policies.
We have not yet found a system that actually suffices. Still, the Jericho Forum did come up with a set of recommendations and issues around the subject such as the need for a powerful, scalable, auditable and flexible system, interoperable global identifiers, the exchange of information access policies linked to the exchange of information and the understanding and governance of information access policies.
The COA framework holds the answer to the recommendations and issues that have been recalled in this section.

2.6.6. Data classification, protection and privacy issues

Introduction:
In this section we observe data classification, data protection, privacy issues and some related topics, based on the study of K. Clark[74], the study of Alina Stan, some position papers, the Open Group Risk Taxonomy and other Jericho Forum related commandments.
Data classification itself is crucial in the perspective of JFC1:

[74] This study is still to be published as we write this section. It will be published and reviewed by the time this study is published as well.

> *"The scope and level of protection should be specific and appropriate to the asset at risk. "*
>
> (Forum 2007a)

If one wants to protect a certain asset, one should apply the proper means of protection. It is of no use for instance to protect your annual report from three years ago by all means available. The yet-to-be-released annual report however does need that level of protection, since the asset is extremely valuable.
The position papers that are used in this section observe the following topics: Issues on data privacy, Enterprise Information and Control systems and Data and Information Management. We finish this section with a set of conclusions. These topics either adhere to the three topics or poses additional value to this section.

Data Classification in the position papers
Little information is found around Data Classification in the position papers. Only some recommendations are provided for:

- **Temporal classification:** as we have seen in the introduction: information can be highly classified for a certain period of time and then change into less valuable or secret, or even become intentionally open to the public. The classification mechanisms should be capable of using a temporal classification in order to handle these kinds of situations.
- **All data must be controllable:** This means that all data should possess access rights and a level of classification, which should continue to work until the data is destroyed. It should even work when spread out over the internet with limitless numbers of copies. The latter is also called "controlling data in the wild".
- **Handling data privacy:** In order to handle data privacy, one should extend the data classification schemes with a personal information category that is further refined with subclasses, to let it handle data privacy issues as stated under JFC10.
- Supporting recommendations for data classification:
  - *Fine-grained information security infrastructure*: One should use a fine-grained information security infrastructure in order to handle all the (rights of the) different information assets.
  - *Using languages such as XACML*: A good candidate language for expressing access right management and the classifications is XACML.
  - *Usage of Meta-information*: In order to save the classification status of an asset, the meta-information should be saved in the body of that asset, together with its classification status. This mechanism can also be used to disallow a reader to read certain classified or top-secret information in an otherwise open document.
  - *Partial classification*: The system should be capable of classifying a document partly. Thus, certain parts are invisible to a limitedly allowed reader, while other parts are readable.
    - *Information classification scheme:* An information classification scheme is needed per organisation as well as universally among the enterprises.

(Forum 2006b; Forum 2007b; Forum 2007e)

Data classification and the study of Kas Clark: Automated Security Classification
We summarise the model and recommendations of Kas Clark made in (Clark 2008), this will be done in short, since there is a large discussion in the Jericho Forum around whether or not the data classification (also called Security Classification) should be automated. If one wants to understand the full details of the model and the appropriate algorithms, then see (Clark 2008).
The model in (Clark 2008) consists of the following elements:

> ***"-Document:*** *Document containing mainly text (rapport, e-mail, etc.) has extrinsic metadata, such as name, author, date created, date modified, size, location, delivery path, etc., and intrinsic metadata that appear in the file contents, such as the subject, title, names of people mentioned, etc..*

-Compliance Policy Repository: *Stores rules regarding personal privacy issues in compliance with government regulation or company policy. (E.g. personally identifiable information, etc.)*
-Compliance Parser: *Parses contents of documents, searching for objects that conflict with the compliance policy. (E.g., credit card information, social security number)*
-Topic Profiles Repository*: Stores profiles for each of the pre-defined document categories. Is created automatically by a machine learning process.*

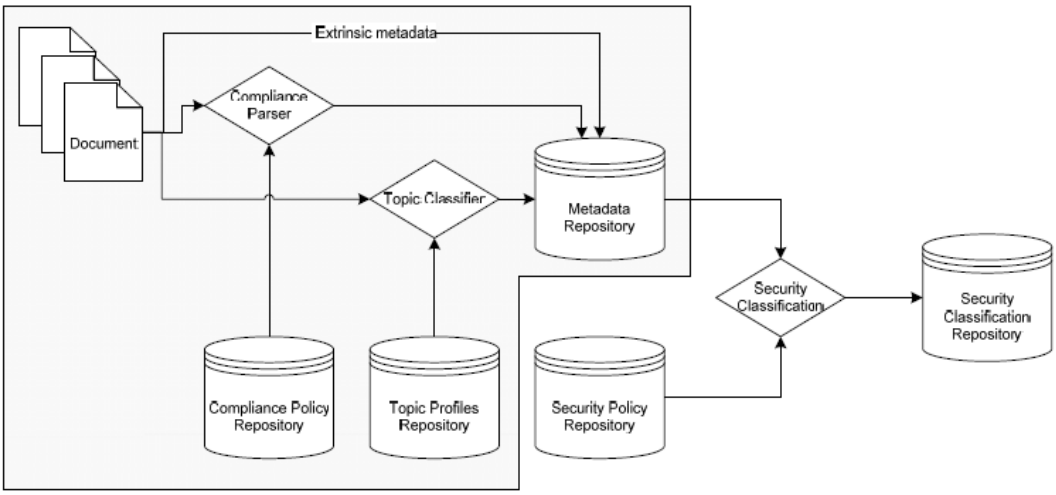


Figure 18: Automated Security Classification model. (Clark 2008)

-Topic Classifier: *Reads contents of document, extracts features and performs statistical analysis, such as Latent Semantic Indexing, Bayesian classification, etc. in order to determine document topic(s). Compares document profile with topic profiles stored in Topic Profile Repository to determine appropriate classification. Stores topic classification in Metadata Repository.*
-Metadata Repository: *Stores output from Topic Classifier module. Can be stored in separate database or in document headers. Contents include intrinsic metadata from documents and extrinsic metadata resulting from content analysis.*
-Security Policy Repository: *Stores rules regarding company security policy used in the Security Classification decision process. Such rules can include classifications based on subjects, departments, authors, age or location (e.g. project X is Top-Secret, author X is unclassified, project X is unclassified if age > 10 years, etc.).*
-Security Classification: *Reads from Metadata and Business Rule repositories. Makes security classification decision based on these inputs. Assigns a security classification to file and stores in Security Classification Repository. Security Classification based on chosen security model (e.g. top-secret, secret, etc.).*
-Security Classification Repository: *Stores security classification of files, as given by Security Classification module. Can be stored in separate database or in file headers. Contents include security classification based on chosen security model (e.g. document X is Top-Secret, etc.).”*

(Clark 2008)

The process of the automated data classification can be summarised as follows:

1. The extrinsic metadata of the documents can be stored in a metadata repository, such as the creation date, date last accessed, return path, relay server, author and location.
2. The compliance parser uses linguistic analysis and tries to find buzzwords, patterns, proper nouns, numbers and uses PCI and SSN algorithms. These data are necessary to match the documents’ content, saved in the compliance policy repository. The findings of the compliance parser on the document are saved in the metadata repository.

3. The topic classifier uses statistical analysis and other algorithms to determine the document's topic and what type of document it is. The findings are saved in the metadata repository.
4. The security classification is given based on the document and the findings inside the metadata repository. Based on what the Security Policy Repository holds in terms of rules, these business rules are company specific. The findings are saved in a security classification repository.

Clark recommends in (Clark 2008) to create a hybrid classifier, which uses the statistical simple classifiers in order to make the easy decisions. Whenever these classifiers are not sure for a threshold of 0.95 or higher, additional classifying algorithms are to be used.

The rationale behind using automated data classifiers is based on the fact that there are multiple successful algorithms and solutions, all focussed on some parts of information classification. If those are combined, a viable and trustworthy automated classification system could be made.
Enterprises can always decide to either depend on them, or use them as assistant classifiers and still let the personnel be in charge of the classification itself. Automated Security Classification will have more speed and consistency and saves considerable amounts of time and money. (Clark 2008)
There are many scenarios in which automated security classification can be used such as auditing, safe storage, Data Leaking Prevention. See (Clark 2008) for more details.

Risk Taxonomy and data classification
The Open Group is currently defining risk taxonomy. As the book around it is still evolving (Fox 2008), we cannot determine the exact impact and value on data classification. However, some things are worth mentioning:

- **Use of common standard around risks (and impact of data loss, etc):** The technical standard described in this book can be used to define the risks and risk levels within the classification system. Systems such as the traffic light protocol (Seccombe 2007) and the inherent risk, could be added to the standard to make a universal data classification standard.
- **Bridge between trust and data classification:** In order to establish trust and to understand the risks of information security threats on certain assets such as documents, the risk taxonomy can be used as a link from data classification to trust.

The traffic light protocol:
An interesting way of classifying, is the traffic light protocol (Seccombe 2007). It groups/defines data in the following classifications:

- **White**: This data can be used in public.
- **Green**: This data concerns 'regular' business.
- **Amber**: this is sensitive data; it can be distributed to groups.
- **Red**: this is highly sensitive data; it can only be distributed to named recipients.

These data classification levels can be linked again to trust levels as seen in (Seccombe 2007). See Figure 19 for more details.

Trust Level	Impact Label	Impact Level $ Measures vary by Company
T5	Catastrophic	Death! End of Company
T4	Material	$250M, Brand
T3	Major	$2.5M
T2	Minor	$25k
T1	Insignificant	
T0	None	

Figure 19: Mapping trust levels and traffic light protocol. (Seccombe 2007))

Data protection: encryption and Digital Right Management System(s)
Data protection is done by encryption and Digital Right Management (DRM) Systems. This thesis does not aim at creating a DRM system or at repeating 'common' knowledge. So, for details around encryption, see (Stan 2008a) and section 2.6.4. We will only observe the recommendations around DRM systems that have been made by the Forum. See subsections "Other recommendations and issues around..."
For now, we understand there should be a flexible DRM System and that multiple ways of digital encoding and encryption can be used. Finally, the JFCs apply here as well. (See appendix 1 and 2)

Other recommendations and issues around Data privacy
The following issues and recommendations have been found about data privacy in (Forum 2007e):

- Issues:
  - *Privacy problems are everywhere:* Privacy problems exist wherever uniquely identifiable data relating to a person or persons is collected and stored, in digital form or otherwise. Improper or non-existent disclosure control can be the root cause of privacy issues
  - *Differences in legal protection around the world:* The legal protection of the right to privacy in general and of data privacy in particular varies significantly around the world.
  - *Difference in ownership:* Much of the data-ownership is allocated to the creator or director of a file structure or a database where the private and/or confidential information is stored. The ownership is not in the hands of the subject of that particular data.
  - *Data mining, fusion and warehousing:* Companies use these techniques to exploit information that has been aggregated from multiple sources. This means the data and thus the amount of Personal Identifiable Information (PII) may vary, outside of the subject's knowledge.
- Recommendations:
  - JFC10 and the full control over the personal information: JFC10 ("Data privacy (and security of any asset of sufficiently high value) requires segregation of duties/privileges.") shows the subject should have the control over his own PII. In fact, the subject should always have the modification, update, restrict and/or destroy rights, even when the container of the data (the file) is copied to another location.
  - *The Enterprise Protection and Control (DRM) of information:* These systems should be extended to also handle data privacy (see the oncoming subsections in this section).
  - *The usage of trust brokers:* Trust brokers should assist in identifying both the subject and the personal information, and subsequently act as a broker to control, spread, exchange and/or release that data.
  - Permission to access/use data when it is held outside of your control:
    - An organisation must take reasonable steps to destroy or permanently de-identify personal information if it is no longer needed for any purpose for which the information is allowed to be used or disclosed. The last two issues are data security issues.
    - End-users should be allowed and be able to demand full disclosure of the use of personal information as part of any transaction.
    - End-users should be allowed and be able to demand that all PII is inherently secure. (JFC9)
    - If privacy protocols are to become adopted as standards, they must be open and interoperable (JFC3).
    - Information privacy or information protection in this context is not about keeping personal information secret. Instead, it is about creating a trusted framework for collecting, exchanging and using personal data.
    - IT and Information Systems need to be adequately secured to prevent unauthorised access to and disclosure of information. One problem with information privacy is the

inevitability of a mistake such as a system's misconfiguration that may lead to exposure of personal information.
  - *The data classification schemes:* As previously stated, the data classification should also support the classification of PII.
  - *More research:* More research should be done in order to create an easy solution to privacy in a de-perimeterised world.
  - *A PII broker:* There should be a PII broker, holding a subject's data, and acting as a single point of reference to the subject's information.
- Other points from the paper:
  - *Legislation:* Privacy matters emphasise four basic privacy principles:
    - Notice – data collectors must inform users of what personal information is collected and who else might share it.
    - Choice – subjects must have the ability to decide how personal information is handled by choosing to opt in or opt out as usage may differ from the original intent.
    - Access – subjects must be able to view data collected about them, and they must have control in correcting inaccuracies.
    - Security – reasonable security must be in place to ensure the accuracy and security of the data.
  - *Platform for Privacy Preferences (P3P):* The Platform for Privacy Preferences (P3P) is best known of the proposed standards and is an XML standard produced by the World Wide Web Consortium (W3C). The P3P project enables Web sites to express their privacy practices in a standard format that can be retrieved automatically and interpreted easily by user agents.
  - *Enterprise Privacy Authorization Language (EPAL):* a formal language designed to specify fine-grained enterprise privacy policies. Unlike P3P, EPAL defines the privacy-practices that are implemented inside an enterprise. The Customer Profile Exchange, or CPExchange, specification defines a data format for disclosing customer data from one party (customer/enterprise) to another. It enables the specification of privacy meta-information as an option and associates privacy controls with subsets of profile information. The privacy meta-information includes the exchange partners, the applicable jurisdiction and a privacy declaration (based on P3P).
  - *RFC 2965 guidelines on state management mechanisms for security and privacy protection*: For users connecting with organisations over the Internet, informed consent should guide the design of systems that use cookies. A user should be able to find out how a web site plans to use information in a cookie and should be able to decide whether those policies are acceptable. In this area, RFC 2965 provides guidelines on state management mechanisms for security and privacy protection when creating stateful sessions with the Hypertext Transfer Protocol (HTTP) i.e. cookies. For privacy, both the user agency and the origin server must assist informed consent**.**
  - *Technologies and countermeasures for privacy:* Two general technology categories have emerged – Privacy-Invasive Technologies (PITs) and Privacy-Enhancing Technologies (PETs). [75]

(Forum 2007e)

Other recommendations and issues around Data/Information management

In the Position Paper 'Data/Information Management' (Forum 2007b), a set of issues and recommendations have been made around data classification and the management of data and information. This paper is in close relation to that around the 'Enterprise Information Protection and Control' (see next subsection). The data classification has already been handled in this section, yet the other issues and recommendations ought to be noticed:

- Issues:

[75] These are out of the scope of this thesis.

- *Scalability of perimeterised information access control methods:* The access control methods in a perimeterised environment will not be capable of scaling to the de-perimeterised environment since the information flows through the corporate boundaries and much more information needs protection.
- *Access control and accountability is too coarse-grained:* the current information security infrastructure is too coarse-grained for a de-perimeterised environment.
- *Compromised "data in the wild" cannot be controlled:* As soon as data is compromised and/or becomes distributed, it is virtually impossible to retain control of it, its exploitation and use.

- Recommendations:
  - *Information protection:* If confidential and integrity-sensitive information could be rendered unusable to anyone who has it in their possession, unless they used the proposed infrastructure to request access, as well as the credentials required to achieve access, it would not matter where your information was or who had access to it. This supports both Information Assurance and Intellectual Property protection.
  - *Legislation:* Legislation declares that if you do not properly protect an information asset, you lose legal rights to it; e.g. if you expose pre-patent information before the patent has been granted, you lose the right to patent it. Maintaining control over your information is crucial from both a business and a legal perspective.
  - *Blanking out:* The information protection system should have the ability to reduce the sensitivity of information by removing pages form a document or blanking out paragraphs of text, allowing a larger document to contain a small section of confidential or integrity-sensitive information to be removed, prior to sharing.
  - *Meta information inside the body:* Meta-information about the document should be a part of the make-up (body) of the information that effectively carries confidential and integrity-based sensitivity classification policy with it at all times.
  - *The infrastructure:* The infrastructure should be capable of supporting all of the recommendations above. Some data should be protected at all times, wherever and whenever (see also Figure 20). The infrastructure that is software-based can add the sensitivity classification policy to the information at the owner's side, then apply a user or role-based access control policy at a remote end-point by comparing user credentials and validating access requests to the access control policy determining an authorization decision. End-point security is crucial; it either has to be strong, or so weak as to be insignificant in the model. The software is required to use open standards and ought to be free from proprietary hardware and software platform constraints, so that anybody can use it within the computing system of their choice. Ideally, it would not always rely on real-time network connection or the information may be unusable offline, which is probably not acceptable for most of the time. This suggests the access control software needing to be lightweight, with minimal applications to be installed on the machines of individuals who intend to utilise the security application, without adding extra security loopholes or concerns to their software environment. Seeing how the access control policy should be stored away from the information resource itself, this may require time-based authentication and authorisation tokens to be created for use offline.

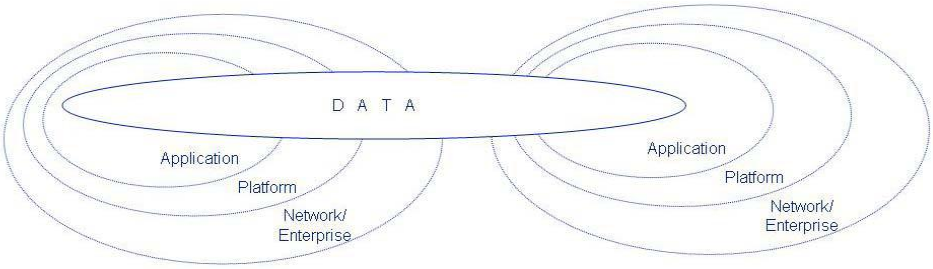


Figure 20: Data centric security. (Forum 2007b)

- *JFC10:* looking at JFC10 (*"Data privacy (and security of any asset of sufficiently high value) requires segregation of duties/privileges."*), it highlights the fact that permissions, keys, privileges et cetera, must fall under independent control. Otherwise, there will always be a 'weakest link' at the top of the chain of trust.
- JFC11: JFC11 is a recommendation in itself: "By default, data must be appropriately secured when stored, in transit and in use".
- *A standard defined for a policy language:* A standard must be defined for a policy definition language, similar to that of XACML. The policy will be applied by the owner of the information and will be bound to the information itself at creation time. A user- or role-based access control policy would be required in order to determine the level of enforcement of an information sensitivity classification policy for each user. It must be stored and managed remotely, being applied at a remote policy decision point upon request for access. This policy will also control resource versioning by being adaptive to access restraint changes/management and enforcing them upon the next request for access. Due to time-consuming overheads, this requirement may only be applied to sensitive information within an enterprise - i.e. that information that requires special handling.

(Forum 2007b)

Other recommendations and issues around Enterprise Information Protection and Control

The last paper we shortly summarise in this section is the Position Paper 'Enterprise Information Protection and Control[76]' (Digital Rights Management) (Forum 2006b). This paper is in close relation with the other two papers that have been summarised. It deals with the following subjects:

> *"• The digital management of rights to access information / data*
> *• Control over that information/data; CSMU (copy, store, move and use) of works*
> *• Integrity of the information/data in question*
> *Information protection needs to cover all data from word processing documents, to data within databases and executable code; it covers enterprise and personal data and is not just confined to protecting entertainment media, which tends to be generically referred to as "Digital Rights Management" (DRM).*
> *Information Control is concerned with the business processes that, for confidentiality, segregation of duties, legal and commercial purposes, tracks rights, rights holders, licenses, sales, agents' royalties and/or associated terms and conditions using the digital technology to*
> *apply and enforce the control. EIP&C does not mandate encryption. Within a secure system data may be unencrypted and other technologies such as hash functions or watermarking may be adequate if tamper protection or proof of ownership rather than confidentiality is required."*
>
> (Forum 2006b)

These issues and recommendations have been found:

- Issues:
  - *Current systems are flawed:* Most of the systems store their data unsecured and rely on network security controls. The lost system with client information, or the DB administrator who has access to all personal information in a database, are examples. Furthermore, there is a data leakage via personal lapses, process imprecision, network file-shares, FTP (File Transfer Protocol), E-mail, USB-Disks, CD burners, et cetera. We cannot lock down the hardware since that generally inhibits business.

[76] From here of referred to as EIP&C in both quotes and text.

- *Need for management:* One needs to be capable of managing/changing/revoking access for data that is outside one's immediate control. Currently this is difficult since a significant amount of information exchange is done between individuals and organisations.
- *The solution is there, yet proprietary:* The current EIP&C solutions offer the ability to provide protection and management at the data layer, irrespective of the location of the data, yet solutions are proprietary, limiting their applications by enterprise domain, operating system family or to specific applications.

- Recommendations:
  - *An open standard for EIP&C:* An open EIP&C standard needs to be developed. This standard should enhance the adoption of EIP&C systems and should allow the tools of different producers to exchange data and information (policies).
  - *An open, inherently secure protocol:* there should be an open, inherently secure protocol (JFC4) for communication between consumers of EIP-/C-protected data and the server or enterprise that controls the data's EIP&C attributes.
  - *Forcing programs and files:* Files should be capable of imbedding an agreed standard set of EIP&C Metadata without the need to know the EIP&C product. Furthermore, programs must also be capable of being forced (probably only in a corporate environment) to input EIP&C Metadata, (example; by a flag in a configuration file, or a setting in the Windows registry) ensuring that entering EIP&C Metadata can be mandated.
  - *Non-EIP&C tool interaction:* documents under EIP&C control must have sufficient clear classification information to ensure that non-EIP&C systems (such as other programs, storage systems etc.) understand how to correctly handle that document. Such classification information must be protected to ensure that tampering with that in-clear information is detectable.
  - *No reliance on network connections*: the EIP&C tools should not rely on a pervasive, ubiquitous real-time network connection (unless this attribute is defined for a particular document) thus enabling off-line working in airplanes, remote places in the world or any other environment where real-time connectivity is not possible.
  - *Every piece of data:* EIP&C tools should manage data in any kind of "data container": varying from discrete files to e-mails and even database records.
  - *Key escrow and key management:* The encryption of documents with a password inevitably leads to documents that cannot be read. The use of EIP&C must enable the management of keys, key escrow and/or access centrally, in such a way that access and functionality can be added/changed/revoked simply and easily. Key management must operate (if allowed by the document EIP&C attributes) in an offline mode.
  - *User Identity Management of users outside of your domain:* the EIP&C tools will have to work with both the corporate as well as the third party-users. That is why it should support the full lifecycle management of those users as part of the EIP&C tools. This automatically means that there should be a federated or similar model for user management.
  - *Security functionality:* A user, as well as his point-of-connection, must be factors to allowing the access to data, depending on the method of accessing/viewing the data, the security of the end-point, and thus the ability to trust the operating environment.
  - *Auditing of Digital Rights information:* Good EIP&C enables audit and thus EIP&C data and access to that data should be capable of being audited. This is especially important when data is being accessed by systems that are outside of the rights managers control such as third party systems, or systems that are off-line when interoperating with the data. The linkage of any EIP&C policy manager to the (corporate) directory should ensure adequate segregation of duties on sensitive data (JFC10).
  - *Control of data "in the wild":* If EIP&C is to deliver a viable corporate system, the data "in the wild" must be controllable, with the ability to effectively destroy that data (void all access), add and/or change and/or extend access and change the EIP&C attributes of the

data. Such controls must support and integrate into the data information management lifecycle, including support for archiving and the retrieval of EIP&C Data from that archive. (Forum 2006b)

Concluding:

After studying the topics of this section, we can conclude the following:

- **Data classification** is about classifying the data to a certain security level in order to understand what level of protection should be applied.
- **Recommendations around data classification:** The following recommendations have been given around data classification: the need for a temporal classification, the need for controllable data, handling data privacy. Furthermore, the need for a fine-grained information security infrastructure, the usage of XACML, the use of meta-information, partial classification and a universal information classification scheme.
- **Automated Security Classification** is a model that focuses on automated data classification, or at least the security classification of it. It consists of multiple elements such as the document that has to be classified, the compliance policy repository and parcer, the topic profile repository and classifier, the metadata repository, the security policy repository and the security classification (repository). All are used to automate the document security classification. This model can be used for multiple purposes and in multiple scenarios, as an aid or as a replacement of human efforts.
- **Automation of the security classification** is still a point of discussion inside the Jericho Forum: should it be (partly) automated or not?
- **Risk taxonomy** is a technical standard for evaluating risk. It helps the classification process as being a bridge between trust and data classification. It also provides a common standard describing risks (which will be inherent to the use/exposure of valuable information).
- **Traffic light protocol** is a protocol for data security level classification. It works with white/none, green, amber and red level data.
- **Data protection**: it is important to use data encryption and Digital Rights Management systems that are open and inherently secure.
- **Issues around data protection:** the following issues have been found around data protection: the scalability of perimeterised information access control methods is inadequate, the access control and accountability structures are too coarse-grained and the currently compromised or “data in the wild” cannot be controlled.
- **Recommendations on data protection:** there are several recommendations like: incorporating metadata such as the sensitivity rating in the body, the capability of blanking out certain parts (and only gaining access to it by authentication and authorisation based on the credentials) and the creation of an infrastructure that would support such actions. Furthermore, recommendations have been given by taking JFC10, JFC11 and XACML in mind.
- **Data privacy** problems exist wherever uniquely identifiable data relating to a person or persons, is collected and stored. There are international differences in legal protection on this matter and differences in ownership and other processes create problems as well.
- **Recommendations on the field of data privacy:** The following recommendations have been stated: JFC10 should be followed, The Enterprise Protection and Digital Right Management systems should take the issues around trust in mind and trust brokers should be used handling privacy issues. Furthermore, a set of recommendations has been given in order explain how/when/why personal identifiable data should be used when it is outside the subject’s control. Yet, more research is necessary to fully control and properly protect PII.
- **Privacy and other developments**: there are multiple developments in current society around digital privacy such as legislation, Platform for Privacy Preferences, RFC 2965 and the Enterprise Privacy Authorisation Language.
- **An open standard for Enterprise Information Protection:** the current information systems are mostly flawed and data leakage occurs easily. Thus, there is a need for better management and the solution is there, yet proprietary. So in order to provide better management, an open EIP&C standard needs to be used, based on open and inherently

secure protocols, which can consecutively be forced on programs and files and can interact with non-EIP&C tools. It should also support all the necessary standards to interact with Identity Management and auditing processes.

COA V2.0 UPDATES:

The following relevant additions have been made in the COA 2.0 revision:

- **Additional EIPC needs**, in: "COA Paper Secure Data: Enterprise Information Protection & Control", which can be summarized as follows:
  - *A standard for handling in-clear classification information:* EIP&C systems must have enough classification information in-clear to ensure that non EIP&C systems understand how to correctly handle that document. The information needs to be stored in such a way that tampering with that in-clear information will be detectable.
  - *The need for an inherently secure protocol for communicating protected data* between the consumers of EIP&C and the server or enterprise that controls the data's EIP&C attributes.
- **Impact sensitivity Categorization for data classification**, in: "COA Service Trust Management: Impact Sensitivity Categorization", which can be summarized as follows:
  - *Five control items for managing the way in which users access and handle data.* The control items consider information creation, storage, sharing, transfer and deletion. See the paper for more details.
  - *Inclusion of the traffic light protocol.* The mapping that is shown at figure 19 is used as a standard in Rev 2.0.
- **Additional details around Data classification**, in: "COA Paper Information Classification", which can be summarized as follows:
  - *Reasons for classification.* The following reasons have been provided: the need for control over access to sensitive or confidential information, protection of sensitive or confidential information and simplifying the discovery of sensitive or confidential information.
  - *Risk to data varies by location.* The classification levels for the data are location independent, however, the risks vary per location. Many more risks are introduced when accessing content from a cybercafé instead of a corporate office.
  - *Issues around consistency.* The Position Paper shows that it will be hard to apply consistent information classification, since the value of the data will be very subjective to the person that classifies it.
  - *The use of automated classifications.* In Rev 2.0 there is more space for automated classification. See the Position Paper for more details.
  - *The use of multiple classifications.* One can use multiple classifications.
  - *The impact of data aggregation*. Data aggregation will impact the classification levels. See the position paper for more details.

The publications can be found at: https://www.opengroup.org/jericho/publications.htm

2.6.7. Identity Management, user authentication and federation

Introduction

In this section, we look at Identity Management, user authentication and federation. The topics are closely related to each other and bound by JFC8 (*"Authentication, authorisation and accountability must interoperate / exchange outside of your locus / area of control."*). We have seen some measures to be capable of authentication and other aspects of Identity Management in paragraph 2.2; however, we focus on the "Jericho Point of View" in this section.

We start with a summary from a keynote that Dick Hart gave on Identity Management; from there on, we observe the laws of identity, Identity Management from the Open Group, the position papers and the different models for Identity Management. After that, we work through authentication and authorisation as described by Jericho, some Identity Management systems and we finish with a summarising conclusion for this section.

Identity Management 1.0, 1.5 and 2.0, a keynote and the Open Group perspective
In 2005, Dick Hardt gave a keynote (Hardt 2005) in which he described the different versions of Identity. We attempt a summary as follows:

- Identity:
  - *Definition*: Identity can be defined as: "Who you are, what you are, what you do, what you like, have and have done, it is what others say about you, which is often more trusted..."”So one could say that identity is reputation".
  - *Conveyed:* Historically it was verbal, once you meet you introduce yourself. Later on came patents of nobility, then official papers for trust on a local scale. After that came the modern identity: photo-ID: driver's license, passport: all for trust on a global scale.
  - *Used in:* Identity is used in identity transactions, there are identity transactions that are:
    - unverified: like a verbal introduction, live or by telephone, and filling out a job application.
    - verified: transactions by means of a passport, which is verified by the government.
  - *Asymmetrical trust*: One does not know the verifying party/the giver of the credential. This creates extra privacy and loads of scalability. Credentials are reusable by any recipient who trusts the issuer.
  - *Modern identity*: The modern identity is a separation between acquisition and presentation, identification and authorisation.
  - *Digital Identity:* mostly known as site registration: unverified and has fewer trust cues.
- **Identity 1.0:** is also seen as the "directory centric"-identity, currently in use. Mostly a username, a password and perhaps a token ID together make up for a directory entry. The trust decision will be opaque. It is under a single authority, the identity is not portable and stored in a silo. A site centric verifiable identity is on the same principle and still Identity 1.0. Identity 1.0 for websites often just means what the site knows about its users. On Figure 21 we see how it works: a user gives account credentials in order to reach a resource. The resource checks with its directory or silo of user credentials whether this user is registered and allowed to access the resource. The more resources there are, the more management there is, the more complex it becomes.

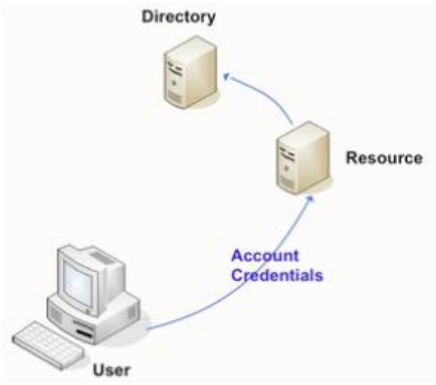


Figure 21: Identity 1.0. (Hardt 2005)

- **Identity 1.5:** Is federation-based and might be used by multiple companies as an easy step forward from identity 1.0.
- **Identity 2.0:** user centric. Works like a digital passport, should be open and simple. It will

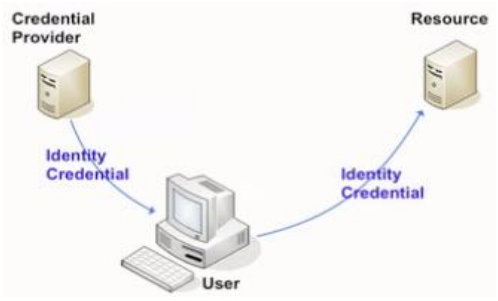


Figure 22: Identity 2.0. (Hardt 2005)

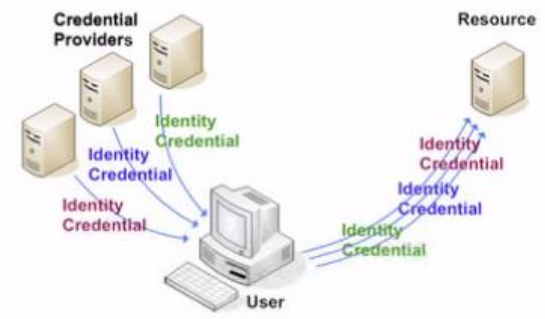


Figure 23: Identity 2.0 with multiple credential providers. (Hardt 2005)

provide transparent policies, simple scalabilities, flexibility. One can use either multiple credential providers or just one, who will provide the user with identity credentials, which can be used to be authorised to a source.

- **Current solutions**: There are multiple solutions: Microsoft .net Passport, Infocard in a "metasystem", Xdi rdi, LID: lightweight identity, OpenID, Passlesl, SAML, OPENSAML, WS-*, SXIP. None of them have been fully adopted, since all of them still have issues or need new investments.

(Hardt 2005)

Kim Cameron's view on identity and the laws of identity:
Kim Cameron, an Identity architect at Microsoft, introduced the seven laws of identity in (Cameron 2005). Before he did so, he first shows the current situation of digital identities, which can be summarised as follows:

- **Internet design:** The Internet was built without a way to know who and what you are connecting to.
- **A patchwork of identity one-offs:** workarounds for the missing identity layer: mostly username/password-based. It is used by many sites and is very insecure since users are bound to insert loads of personal information and cannot evaluate afterwards if the parties are legitimate or if the site is authentic.
- **Criminalisation of the Internet:** Phishing and Pharming by spyware attacks are increasing tenfold yearly. If nothing is done about the situation, a deep public crisis due to the accumulating threats against the unsafe ad hoc nature of Internet identity, will be evident.
- **Integration (and web services) unusable:** As long as there is no integrated solution or an Identity Management layer to the internet, many integration services and web services are unusable since they will break privacy or will be prone to many threats.
- **Adding an identity layer will be hard:** There have been local successful attempts at creating an identity service to the internet; none have been universally successful due to the lack of agreement between nations (legislation and more), enterprises or financial sectors due to multiple reasons.
- **Simplistic digital identity solution is not realistic:** due to the many threats and the lack of agreement among the users and other parties, a panacea in the form of a simplistic and single digital identity solution is not realistic.
- **A unifying identity metasystem:** In order to create an identity layer for the internet that everybody can use, one needs a metasystem in which all of the other solutions can be integrated, so that it can be accepted globally and within all of the different sectors of the market.

(Cameron 2005)

He has the following definition of a digital identity:

> *"A set of claims made by one digital subject about itself or another digital subject"*
>
> (Cameron 2005)

A subject is defined as:

> *"…a person or thing represented or existing in the digital realm which is being described or dealt with".*
>
> (Cameron 2005)

This "dealing with" can be anything: humans, devices and computers, digital resources, policies and relationships between other digital subjects.
A claim is defined as:

> *"…an assertion of the truth of something, typically one which is disputed or in doubt".*

(Cameron 2005)

Claims are for example: A student number provides information as to that the subject knows a given key, it may function as a conveyor of personal information (name, address, date of birth and citizenship), or ascertain whether that someone is part of a certain group (age above 16) or that a certain capability is given.

The following Laws of Identity have been defined (again, in (Cameron 2005)):

1. **User Control and Consent:** Technical identity systems must only reveal information identifying a user with the user's consent.
2. **Minimal Disclosure for a Constrained Use:** the solution that discloses the least amount of identifying information and best limits its use is the most stable long-term solution.
3. **Justifiable Parties:** Digital identity systems must be designed so the disclosure of identifying information is limited to parties having a necessary and justifiable place in a given identity relationship.
4. **Directed Identity:** A universal identity system must support both "omni-direcitonal" identifiers for use by public entities and "unidirectional" identifiers for use by private entities, thus facilitating discovery while preventing unnecessary release of correlation handles.
5. **Pluralism of Operators and Technologies:** A universal identity system must channel and enable the inter-working of multiple identity technologies run by multiple identity providers.
6. **Human Integration:** The universal identity metasystem must define the human user to be a component of the distributed system, integrated through unambiguous human-machine communication mechanisms, offering protection against identity attacks.
7. **Consistent Experience Across Contexts:** The unifying identity metasystem must guarantee its users a simple, consistent experience while enabling separation of contexts through multiple operators and technologies.

See for more details: (Cameron 2005).

Identity Management from the Open Group perspective

The Open Group has created a whitepaper in collaboration with Skip Slone (Area 2004) in which they describe their view and give their definitions on the subject. This whitepaper can be summarised as follows:

- Identity:
  - *Defintion1*: "Identity is defined as the quality or condition of being the same; absolute or essential sameness; oneness. Identity is what makes something or someone the same today as it, she, or he was yesterday. Importantly, identity can refer to a thing (e.g., a computer) as well as a person. Things and people can have different identities when working with different systems, or can have more than one identity when working with a single system, perhaps when working in different roles."
  - *Definition2*: "Identity is the fundamental concept of uniquely identifying an object (person, computer, etc.) within a context. That context might be local (within a department), corporate (within an enterprise), national (within the bounds of a country), global (all such object instances on the planet), and possibly universal (extensible to environments not yet known). Many identities exist for local, corporate, and national domains. Some globally unique identifiers exist for technical environments, often computer-generated."
  - *Insufficient:* Many identities of today are insufficient for business goals. For instance, a social security number is not unique and is not complete for identifying a person.
  - *Attributes and identifiers:* a person can have a single identifier, or multiple (name and social security number). Other things can be added as well such as relationships (such as

bank accounts), affiliations, role-based rules, roles, temporary privileges, multiple profiles (such as personal, professional, consumer).

- Identity Management:
  - *Definition*: "Identity Management (IdM) is a convergence of technologies and business processes. There is no single approach to Identity Management because the strategy must reflect specific requirements within the business and technology context of each organisation."
  - *Drivers:* enablement of higher level of e-business, reduction complexity of integrating business applications, management of flow of users entering, using and leaving the organisation, support global schemas for certain categories of operational tasks, a need for a response to the pressure from the growing numbers of Web-based business applications that need more integration for activities such as single sign-on.
  - *Verification:* Verification is the process of establishing identity prior to the creation of an account that can later be used as an assertion of identity. Requirements for verification are generally based on the sensitivities of the identity itself.
  - *Authentication*: Authentication is the process of gaining confidence in a claimed identity. Once identities are issued, whenever they are used, there is the requirement that the person using the identity is the person that is qualified to use it. Authentication usually happens in a process with an authentication authority.
  - *Revocation*: Revocation is the process of rescinding an identity that has been granted. This process must be properly recorded for audit purposes. All systems and processes with which identity has been established must now be notified that identity was revoked. If it is wrongly implemented, then there are potential significant liabilities.
  - *Provisioning and authorities*: the key aspects of provisioning:
    - Account provisioning, which deals with identity-related information associated with individuals, their personal attributes, affiliations, et cetera. It is one of the core functions that may be performed during an identity's lifecycle. Other functions in account provisioning are adding, modifying, suspending and resuming an identity.
    - Resource provisioning, which deals with business assets such as computers, databases, and applications and the management of permissions associated with those assets
    - Account de-provisioning, which deals with the termination of access rights to systems and services and re-allocation of those systems and services
    - Trust: All provisioning must be based on the concept of a trusted identity. Trust in the identity can be established by either contact or some other means.
    - Additional approval and delegated administration: one should use the concept of delegated administration and, if necessary, multiple additional approvals before an identity is granted access to certain systems and services.
  - *Multiple sources:* There are often multiple authoritative sources in an organisation; those should be the main source of identity information.
  - *Permission management and authorisation:* permissions that need to be managed are the permissions to access, compare, write, modify, create, destroy, execute, copy, print, forward, delegate, purchase, authorize, approve, sell, sublease, assign, transfer, hire, fire, promote, and so forth. They are not attributes of an identity, yet they some can be derived from it as such. The source of authority for the permission will vary depending on the source where the permission is about. Access controls can be used to enforce the permission management decisions by access control lists, attribute certificates and digital rights management. In order to authorize an individual, permission needs to be allocated to him. This has to be done by an authorization authority.
  - *Directories and their roles:* Identity Management should be seen as a ubiquitous, interoperable facility is in its very early stages of development. As it matures, the use of the directory as part of its inherent infrastructure will also grow and mature. The following issues/ recommendations have been made:
    - The directory structure has been Identity Management favoured, yet it could be unstable from time to time due to very dynamic information around wireless devices.

The directory services underlying the Identity Management becomes ever more important because the information has to be published and protected. Currently only the X.500 protocol has a standard for access controls, LDAP (Lightweight Directory Access Protocol) and alike have no such thing. This becomes very important whenever organisations want to exchange information around identities in order to collaborate.
    - Another directory service issue in collaboration is interoperability of the Identity Management formats and directory services. This seems partly resolved in different solutions, yet it would need more maturity according to the paper. Using Identity Management standards should speed up this process.
    - All directory services should have decision-making logic, like the Network Operating System Directories such as Active Directory and e-directory. Policy matching should be added as well in order to create the facilities for federated identity.
    - The general-purpose directories do not function as "enforcers" of the policies, yet X.500 directories do. Other solutions that are capable of doing so are mostly proprietary.
- **Relationship to trust:** In order to trust someone, we need to be capable of relying on that other person. That other person has to be trustworthy. The identity is closely related to that.

(Area 2004)

The rest of the whitepaper covers the business rationale, the security point of view on Identity Management and a further explanation of the terms used in this subsection. That is outside the scope of this thesis for now. See (Area 2004) for more details.

Identity Management, user authentication and federation according to the Position papers
In our study on this subject we found one position paper around the federated identity (Forum 2006c). The message of the paper can be summarised as follows:

- **Problems and issues:** the idea of the Federated identity approach will require one organisation, the Identity Provider, to be in a privileged position in control of the issuance and/or validation of credentials. Many companies do not want to pass control of a major asset to another entity. Furthermore, the solutions often have asymmetrical trust issues and sometimes even privacy issues. Many approaches have also been limited to authenticating human users instead of their resources as well. Finally yet importantly, these federated identity technologies do not directly match the key business needs and trust relationships for a de-perimeterised environment.
- **Recommendations:** The following recommendations have been given:
    - *Privacy concerns must be visibly met*. One of the measures to do so is clearly distinguishing between credentials and attributes. Shared secret credentials should in general not be transferred to other organisations, due to the increased risk of compromise.
    - *Simple and strong:* a solution should be given that allows users to have a simpler and stronger way to authenticate to organisations or between organisations.
    - *Peer-to-peer*: there should be support for peer-to-peer authentication and for a separate identity provider. Organisations should also be allowed to work as peer-to-peer nodes.
    - *Interoperability and N credentials/ N technologies*: There should be multiple interoperable authentication technologies and different credentials referring to the same individual. Depending on the context, one of them should be used. Any system should be flexible and extensible.
    - *Data attributes held by end-user*: In most cases data attributes should be held by the end-user, rather than centrally stored by a third party. For browser-based applications, a standardised data form schema would make it simple to pass the same data to different organisations completely under user control. Individuals should be able to choose which sets of attributes are used for a given transaction (work/home address, credit card selection).
    - "Challenges to the industry":

- Create common schemas for the majority of transaction data attributes requested, including name, address and payment details, to remove the need for centralized attribute storage.
- Mutual authentication should be used by default.
- Peer-to-peer authentication should be permitted, without the need of a privileged identity provider.
- The currently assumed role of an individual should be made explicit to systems.
- Subject attributes should not be used as credentials.
- Credentials and authorisation information should be able to be transferred between organisations using open protocols and standards, and be simple to manage the equivalence relationships.
- It should be possible to support a multiplicity of credentials and technologies for an individual.

(Forum 2006c)

COA V2.0 UPDATES:

The following relevant additions have been made in the COA 2.0 revision:

- **Relevant Identity Management aspects in a de-perimeterised environment**, in: "Identity Management – Federated Identity", which can be summarized as follows:
  - *Federated Identity Management leads to potentical privacy issues*: the usage of the combination of user attributes and user credentials could cause legal problems and give rise to privacy issues.

The publications can be found at: https://www.opengroup.org/jericho/publications.htm

Identity Management, user authentication and federation according to the Jericho in depth works:

In this subsection, we look at all of the recommendations and issues that have been mentioned around these topics in the Jericho in Depth series. We will focus on (Barannikov 2008; Teheux 2008) for now.

The following definitions have been found in these works around Identity Management:

- **Identity Management** is a broad administrative area that deals with identifying individuals in a system (such as a country, a network, or an enterprise) and controlling their access to resources within that system by associating user rights and restrictions with the established identity. (Barannikov 2008)
- **Identity Management** is an integrated system of business processes, policies and technologies that enable organisations to facilitate and control their users' access to critical online applications and resources – while protecting confidential personal and business information from unauthorised users. (Barannikov 2008)
- **Identity** is the set of characteristics that somebody recognizes as belonging uniquely to himself or herself and constituting his or her individual personality in life.
- **Digital identity:** A digital identity contains data that uniquely describes a person or a thing but also contains information about the subject's relationship to other entities.
- Three Tiers of Identity:
  - Tier 3: abstract Identity: abstract data such as demographics, it says nothing about a person self.
  - Tier 2: shared identity: attributes/preferences / traits. May be assigned by others or acquainted in the course of time. Such as a job or a role.
  - Tier 1: personal identity: Unique information / traits. This is something unique and in possession only by a single entity.
- **Identity Lifecycle**: the combination of Identity Management processes

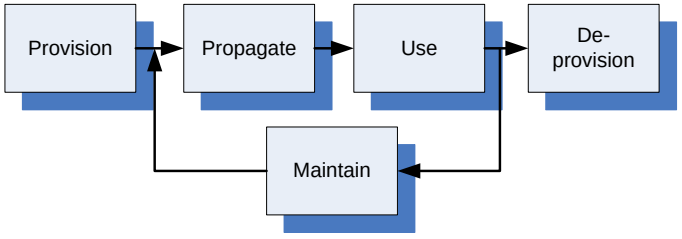


Figure 24: Digital Identity Lifecycle (based on (Barannikov 2008)).

constitute the Identity Lifecycle:

- *Provisioning:* the process of providing a new identity. This is mostly an account with attributes related to the identity.
- *Propagating:* after the record (the identity) is created, it is propagated to other identity resources that are suitable for Identity Management control. It will be reproduced through secure protocols in all systems in the whole infrastructure.
- *Using:* the user uses his digital identity to gain access to resources. Using is the only process that may extend outside the boundaries of the local domain.
- *Maintaining:* the attributes of a user may change in the course of time. Identity data should reflect those changes and provide up-to-date information to the requester. The most changing attribute is the password.
- *Deprovisioning:* when identity is not needed it is removed from the central repository and consequently from all the systems on the network

The following problems, solutions and requirements have been found around Identity Management in these works

- **Identity Management evolution:** The following models of identity have been found and discussed:
  - *Isolated*: a user uses a stand-alone station. He logs onto the system, the credentials are checked against the local user database and the user is authenticated. Data sharing is complicated and Identity Management is very cumbersome. It is completely unsuitable for the Jericho concept.
  - *Centralised*: there is a central directory with all the identities stored. A user request data from a resource, the resource sends an authentication request to the directory server, directory server sends a reply with information about the directory entry and the user is authenticated. The scalability is very limited: identities and policy management are bound to a single domain. Trusts between domains can extend the scalability to some degree. Yet there is a lack for support of multiple systems. Third party software is often needed in order to provide environmental integrity and seamless user experience. There are many issues around the interdomain trust such as vendor locking, interoperability problems and the establishment of single or two-way trust. The limitations are obvious for collaboration. Yet, the centralised model qualifies for a de-perimeterised concept up to the certain degree with help of trust one can achieve communication with other domains but when extensive collaboration is required it fails to fulfil its mission.
  - *Federated*: the definition used here, by the Burton group: “the agreements, standards, and technologies that make identities and entitlements portable across autonomous domains”. The process is as follows: Company A establishes a federation agreement with company B, the user receives his digital identity from company A (identity provider), a digital identity is presented to the company B (relying party) in order to access their resources, (optional :) relying party may verify the identity from the identity provider. There are three types of federation:
    - Ad-hoc federation: two enterprises agree upon unilateral or bilateral federation of the digital identities.
    - Hub-and-spoke federation: a federation island formed around a company.
    - Service provider: a single relying party and multiple identity providers. The parties have agreed upon the unilateral identity acceptance by the service provider. If there is a bilateral federation agreement among multiple providers, then a multi-provider cross-domain can be established.
    - Identity federation network: an organised identity environment where organisations and individuals can freely interact and collaborate. It is privately owned or managed by a non-profit organisation.
  - *With identity federation, one will have to manage accounts and the federation agreement.* Federation can be seen as a passive extension of the identity data outside the boundary, thus an extension to the centralised model. This means that no trust is involved and there will be many risks.

- *User centric*: Multiple identity providers verify the claim assertions by authenticating user or underlying claims. After the claim is verified, the identity provider issues a security token that is presented to the relying parties. This model allows privacy in the digital world. The authentication authority never communicates directly with the relying party. Trust is very important in this model. In order to verify the validity of security tokens, a relying party has to trust the identity provider and should possess the means to authenticate it. There is a lot more managed in this model:
  - The identity provider manages user identification, identity provisioning, identity maintenance, user authentication, identity revocation, trust management.
  - The user manages the identity provisioning, identity maintenance, identity revocation and user authentication.
  - The relying party manages the trusted identity issuers, the security token verification, and the security and authorisation policies.
  - This model truly erases boundaries as envisioned by the Jericho Forum. Users do not belong to any domain any more. The user centric model requires the gradual change of the current infrastructure. The best possible moment for the implementation of this model will be after a company implements identity federation. Another option would be implementing it for usage with the relying parties first, then gradually dissolve the local domain boundary.

(Barannikov 2008; Teheux 2008)

The following definitions have been found around authentication:

- **Authentication:** The process of establishing an Identity to be used in a particular instance, by verifying an assertion.
- **Multifactor authentication:** an authentication process in which multiple credential variations are used, the three factors of the authentication are: something you know (Memometrics and cognometrics), something you have, something you are (biometrics)
- **Network devices authentication:** there are several possibilities for network device authentication:
  - MAC address: the Mac Access Control address is unique and can be used for device authentication.
  - Trusted Platform Module: this is a hardware chip that allows device authentication. It contains unique information that would allow secure communications.
  - Software based device authentication. A Windows Security Identifier, installed software or a license key can create a unique signature that could identify a system on the network.
  - Certificate-based: PKI based infrastructure can be used as such.
  - IP address based authentication: every device on the network has a unique IP address, which allows layer 3 communication with other devices.
- **Related to trust:** Authentication heavily depends on trust. The relying party will have to trust the user that authenticates himself by some or any means.
- **Authentication models:** there are three authentication models: hierarchical (as the current DNS model), flat or hybrid. See (Barannikov 2008) for more details.

(Barannikov 2008; Teheux 2008)

The following problems, solutions and requirements have been found around authentication in these works

- **Requirements:** the following requirements have been found in the named literature:
  - *Federation and User centric:* Support for federation or user centric design.
  - *Practicality*: it should be non-intrusive and easy to use.
  - *Appropriate level of security*: it should be able to re-assert itself if authentication level is insufficient.
  - *Locational transparency*: user must be able to authenticate himself regardless his physical location. Location parameters might be needed for authorisation purposes.

- *Protocol insensitivity*: systems should be able to interoperate regardless the transport protocols they use.
- *Appropriate level of privacy*: authentication system must comply to the privacy regulations of the company.
- *Reliability*: authentication system is a crucial and should be reliable.
- *Auditability*: all transactions in the authentication system should be audited and retained for a required period.
- *Manageability*: user accounts should be easily manageable.
- *Support for multifactor authentication:* additional factors for verification may be required according to policies.
- *Device authentication*: some policies require the device to be authenticated as well.
- *Additional parameters:* authenticating authority must be able to include additional parameters such as GPS data, security status et cetera.

- **Recommendations:** The following recommendations have been found in the named literature:
  - *Biometrics:* Do not use biometrics on a global scale: once it is compromised, it cannot be restored.
  - *Trusted Platform Module and certificates* are the greater potentials for device authentication.
  - *Use of trust architecture:* The trust architecture should be used.
  - *SAMLS* and *Microsoft Identity Metasystem* are both recommended to use for authentication solutions.
  - The Jericho Forum Commandments and Authentication:
    - JFC1[77]*:* it is easier to protect an asset by moving the protection closer.
    - JFC2*:* One should remove the excessive complexity and introduce a simple easily comprehensible security architecture. Security mechanisms must be scalable and must span all tiers of the network
    - JFC3*:* Make sure that you choose a solution that fits to the environment.
    - JFC4*:* One should use simple, open and secure protocols for communication.
    - JFC8: AAA framework must be able to extend outside the security domain. Assertions issued by one authority must be accepted by another, thus eliminating the creation of multiple identity instances.
    - JFC9: Data should be able to protect itself. Security attributes are inseparable from data, but they must be transferable between the domains. Access and access rights have a temporal component.

(Barannikov 2008; Teheux 2008)

Authorisation has been defined as an object of the AAA-framework (Authentication, Authorisation and Accounting). In which authorization refers to the process of making decisions regarding actions to be allowed or denied, based upon information received from other sources. Another definition that we found: *"what the identity can do, in a given instance, as a result of proving an assertion."* (Barannikov 2008). It is seen as the linking pin between all other processes. Information gathered by other processes serves the single goal of allowing the authorization process to make appropriate decisions. (Teheux 2008)
The following problems, solutions and requirements have been found around authorisation in these works:

- **Requirements:** The following requirements have been found:
  - *Authenticate all:* All entities involved in the authorization process should be authenticated;
  - *Rights in the data itself:* The authorization process should use rights stored on the data itself to determine applicable rights;

[77] See for a complete ouline of the JFCs (Jericho Forum Commandments) Appendices A1 and A2.

- *Outside the domain:* The authorization process must be able to handle authentication information obtained from outside the local domain;
- *Robust:* The authorization process must be robust and require the least amount of administrative effort possible;
- *Trust relations:* The authorization process should be able to include trust relations;
- *Secure protocols and standards:* The authorization process should use open and secure protocols and standards.

- **Logical solutions**: the following logical solutions have been found and/or recommended:
  - *Claims-based authorisation:* this is intricately linked with the user centric identity model. Within this architecture, information can require certain claims to be provided before access is allowed. The authorization process determines so. It will also determine what claims are required, which are received and if the received claims are trusted enough to be allowed to enter the final authorization of requests. There is a strong need for user consent: users must be able to determine what claims are required. They also must be able to select claims they will allow the requester of claims to know. Identity Providers provide the user with claims that can be presented to the Relying Parties.
    - Advantages: allows users to control their personal information, allows information to strictly control access.
    - Disadvantages: requires a complex architecture, requires a universal implementation, difficult troubleshooting and management, and requires the development of new solutions.
  - *Passive authorization:* the authorization process is unable to poll processes by itself. The user will give it its credentials; the process checks it against an access control list and allows or prohibits access.
    - Advantages: simple implementation, relatively easy troubleshooting and management, proven technology
    - Disadvantage: lack of flexibility.
  - *Active authorization*: the process is able to actively poll other processes and request additional information if necessary. This can involve extra factors for authentication and the involvement of end-point security checks.
    - Advantages: flexible, more security possibilities, allows integration with the end-point security process.
    - Disadvantages: requires a complex architecture, difficult to manage and to troubleshoot.
  - *Recommended solution:* until there are good implementations of the claims based authorisation, the active authorisation is seen as the recommended solution.
- Process interactions and scenarios:
  - *Scenarios:* There are three scenarios found for the authorisation process:
    - Data access with matching Access levels;
    - Data access with insufficient user Access level;
    - Data access without sufficient authorization;
  - See (Teheux 2008) for more details around the scenarios.
  - *Process interactions:* The following interactions have been found:
    - Input: the process will retrieve input from the authentication process (access level, credentials), the data classification process (data rights), the encryption process (encryption options to be applied)
    - Output: the process will output the following: a listing of actions that can be auditable for accounting, a request for encryption options to the software agents.
- **Technical solutions:** the vendors are currently incorporating authorisation processes in their software. The processes can be used neither in isolation, nor in collaboration with completely different suites. Furthermore, not all of the Jericho Forum Commandments can be followed with the solutions of today. That is why there are no recommended technical solutions.

(Barannikov 2008; Teheux 2008)

Additional: Accounting

As an addition and still in line with JFC8, we should consider accounting as well. We have summarised to contents found in (Barannikov 2008) on this subject:

- **Accounting** is the process of keeping track of online user activity. Accounting data is used for network performance analysis, capacity planning, financial matters, auditing and many other purposes. See also other sections in this paragraph.
- **Auditing data exchange**: Auditing data should be exchanged between the security domains, as JFC8 states. This should allow auditor to see whether a user has accessed resources in other security domains.

(Barannikov 2008; Teheux 2008)

Identity Management systems:

The following Identity Management systems have been described in (Barannikov 2008):

- **SAML**: Security Assertions Markup Language. It is created by OASIS. SAML v.20 has been released in 2005. See for the authentication process (Barannikov 2008).
  - *Components:* SAML consists of several components:
    - Assertions: packages of information that supplies one or more statements made by a SAML authority: authentication, attributes and authorisation decisions
    - Protocols: different request/response protocols are defined: authentication request, single logout protocol, assertion query and request, artefact resolution, name identifier management and name identifier-mapping protocols.
    - Bindings: they define how SMAL messages can be carried in underlying transport protocols.
    - Profiles: they define how SAML assertions, protocols and bindings can be combined in order to achieve a greater interoperability in certain usage scenarios
  - *Short review:* SAML is a complex system and a standard that was adopted by many enterprises. The biggest disadvantages of SAML are its complexity and incompatibility between different versions. Despite this SAML has managed to become de facto standard in identity federation.
- **Liberty Alliance**: a consortium that is formed to establish open standards and best practices for identity federation. They have three important specifications published:
  - *specifications:* The following specifications have been published:
    - Identity Federation Framework: consists of the core specifications that make the creation of the multivendor identity federation network possible. It has been contributed to OASIS for the development of SAML 2.0. It has features such as identity/account linkage, simplified sign on, and simple session management.
    - Identity Web Services Framework: is a general framework for discovery and invocation of identity services. The specifications provide the framework for building interoperable identity services, permission based attribute sharing, identity service description and discovery.
    - Identity Services Interface Specification: describes how a service that supports identity information of a principal self should function.
  - *Short review:* The Liberty Alliance provides standards by adopting other standards and delivering own proposals. Compliance with the abovementioned standards is advantageous for every company trying to federate the users' identities. Their identity federation specifications extend the possibilities of SAML and provide services that SAML lacks.
- **WS-Federation**: SOAP clients and web services can use it directly. It operates with the variety of the security token services and it is heavily dependable from WS-Trust and WS-Security Policy.
  - *Short review:* it exists for a long time, although it has not found wide acceptance among organisations because it is heavily dependant on other WS-* protocols. It offers similar possibilities as SAML 2.0; it still misses some important features like for example broad authentication context.

- **OpenID**: is an open, decentralised free framework for user centric digital identity. A user authenticates himself at his identity provider, which can be a blog or a user home page. A URI is used as an identifier. Principal information and attributes are exchanged between identity provider and service provider/ relying party with user consent. Organisations require an infrastructure with more control and strict security policy. In case an enterprise would be interested to provide its employees with OpenID identities, it would have to create a personal page for every user.
  - *Short review:* the advantages of Open ID are its simplicity and lightweight trust model. The biggest disadvantage is in the security. OpenID is an open source protocol and recently multiple security flaws were discovered.
- **Microsoft Identity metasystem:** it is claimed to use open standards and incorporate multiple protocols that make interoperability between multiple standards possible.
  - *Components:* The following components are part of the MS Identity metasystem:
    - A user agent, which is Microsoft Cardspace, formerly Infocards. It is a piece of client software that enables users to provide their digital identity to online services in a simple, secure and trusted way.
    - Identity Provider or Security token services are the identity providers that supply user with authentication token. Kerberos and X.509 security tokens are supported.
    - Languages that make the conversation between user agent, identity provider and the relying party possible. The languages are often the WS*- languages.
  - *Short review:* Microsoft Identity Metasystem is complex system that delivers user centric experience for the end-users. Though it is based on Microsoft's own standards, it intends to achieve interoperability with other systems.

(Barannikov 2008; Teheux 2008)

Concluding:

Looking back at this section, we can summarise it and conclude the following:

- **Identity** is the fundamental concept of uniquely identifying an object (person/computer) within a context. Since there are multiple contexts, an object can have multiple identities. The object can have or be related to multiple attributes and identifiers. An identity can be (un-)verified. There are multiple definitions around identity, they depend on the context used.
- **Three tiers of identity:** the abstract, the shared and the personal identity.
- **Identity Management** consists of technologies and business processes. It is aimed at managing the identities inside and around the company. The approach of Identity Management is depending on the requirements that the organisation has set. There are several drivers such as reducing complexity, management of flow of users, better integration of applications et cetera.
  - *Laws of identity:* There are seven laws of identity to consider: internet design, a patchwork of identity one-offs, criminalization of the internet, integration (and web services) unusable, adding an identity layer will be hard, a unifying identity metasystem will be required.
  - *Identity Lifecycle:* the lifecycle consists of provisioning, propagating, using, deprovisioning and maintaining.
  - *Identity 1.0* is a "directory centric"-Identity Management concept. It is currently in use. It is complex, uneasy to scale and very difficult to use if one needs to manage cross-organisational identities. It is also known as the centralised identity, the follow up of the isolated identity and it will fail to be usable in intensive cross-organisational collaboration.
  - *Identity 1.5* is also known as "federated" identity. Enterprises can form a federation and can manage and verify identities in collaboration. The problem with this type of approach is that there will be an Identity Provider required, which will have a privileged position in control of the issuance and/or validation of credentials.
  - *Identity 2.0* is also known as the "user centric" identity. A user can use identity credentials form credential provider(s) to authenticate to a certain resource.

- *Solutions:* There are multiple solutions, which have been investigated, SAML, the trust broker architecture and the Microsoft Identity Metasystem are the promising solutions
- *Infrastructures:* It is considered very important to use directory services as a major part of the infrastructure, which will need to mature in multiple ways and within multiple aspects.

- **Authentication** is the process of gaining confidence in a claimed identity. One has to verify that the person using the identity is the person to whom the identity belongs. This can be done with multiple factors. Authentication can also be focused on device authentication.
- **Verification** establishing identity prior to the creation of an account that can later be used as an assertion of identity.
- **Revocation** is the process of rescinding an identity that has been granted.
- **Provisioning** consists of multiple key aspects: account provisioning (in which verification can take place), resource provisioning, account de-provisioning (or revocation), trust, additional approval and the concept of delegated administration.
- **Authorisation** is the process of making decisions regarding actions to be allowed or denied, based upon information received from other sources.
- **Permission management** is the process of managing all the permissions that an identity can have. Different authorities, based on the context, the object, the permission or the permission itself that has to be altered, can do it.
- **Accounting** is the process of keeping track of online user activity. Accounting data is used for network performance analysis, capacity planning, financial matters, auditing and many other purposes.
- **Relation with trust:** Identity Management, authentication, authorisation and accounting have a direct relationship with trust. Most of the processes are based on or trying to increase trust. The trust relationship can be symmetrical and asymmetrical, depending on the implementation.
- Recommendations for Identity Management, authorisation and authentication: the following recommendations have been provided:
  - *User centric and data attributes should be held by end-user:* the studies that have been examined recommend a user centric approach in Identity Management. They also would like to see that the data attributes around and related to the identity should be held by the end-user.
  - *Privacy concerns must be met*: one can do so, by clearly distinguishing between credentials and attributes.
  - *Simple and strong:* the solution should allow users to have a simpler and stronger way to authenticate to organisations or between organisations.
  - *Peer-to-peer* modes in authentication should be supported.
  - *Interoperability* of the Identity Management solutions should be guaranteed. Multiple credentials and technologies should be usable to refer to the same identity.
  - *Development of directory services*: the infrastructure will be directory based, which needs to be further developed.
  - *Mutual authentication*: the authentication must be mutual: both resource and user should authenticate themselves.
  - *Open and inherently secure models and protocols:* there should be a set of common schemes, open and inherently secure models and protocols for Identity Management. At best, the Identity Management should be protocol insensitive.
  - *Authenticate users and devices:* Both the user and the device should always be authenticated together.
  - *No biometrics:* Do not use biometrics.
  - *Other requirements/recommendations:* the solution should be practical, appropriate in terms of security and privacy, reliable, auditable, manageable, have support for multifactor authentication.
  - *Use active authorisation*, until the claims based authorisation works.

2.6.8. Trust, Trustmanagement and Trust brokers

Introduction
As we have seen in section 2.5.5, trust is a very important concept in collaboration. We have also seen that it takes an important place in Identity Management (section 2.6.7). JFC 6 (*"All people, processes, technology must have declared and transparent levels of trust for any transaction to take place."*) and JFC 7 ("*Mutual trust assurance levels must be determinable.")* are about trust as well. Furthermore, it can be related to IT-auditing (establishing and checking trust, see section 2.6.10), End-point Security (establishing and checking trust, see section 2.6.9) and many more concepts. This makes trust a very important subject to look at.
In this section we will do so, by discussing the following subjects: First we will look at what trust is, as defined by the study of Fabian van der Leijden(Leijden 2008), second we will take a look on what trust is based on the study of Andor Demarteau (Demarteau 2008). From there on we will take a look on the Risk taxonomy (Fox 2008) from the Open Group, the Trust broker framework (Bruning 2008a) and the Trust broker services (Bruning 2008b) as defined by the Jericho in depth series.

An important remark should be made: the documentation around Risk taxonomy and the study from Fabian van der Leijden is still in concept as we are writing this section.

What is trust, based on the study of Fabian van der Leijden?
In (Leijden 2008) different definitions of trust have been given. Since this study is about the COA framework and not about trust itself, we will discuss some of the most important definitions and summarise the rest:

The first is from Gambetta in (Gambetta 1988):

> *"trust (or, symmetrically, distrust) is a particular level of the subjective probability with which an agent assesses that another agent or group of agents will perform a particular action, both before he can monitor such action (or independently of his capacity ever to be able to monitor it) and in a context in which it affects his own action…""… When we say we trust someone or that someone is trustworthy, we implicitly mean that the probability that he will perform an action that is beneficial or at least not detrimental to us is high enough for us to consider engaging in some form of cooperation with him. Correspondingly, when we say that someone is untrustworthy, we imply that that probability is low enough for us to refrain from doing so."*
>
> (Gambetta 1988)

In other words, Gambetta links trust to an estimation of the type of actions that others will perform. In (Nooteboom 2005) we found some more details around the concepts, most of it, speaks for itself:

> *"Georg Simmel proposed that trust is a mixture of rationality and feeling: it is based on certain amount of rational assessment, but also entails a leap of faith beyond that. This seems related to the 'paradox of information' associated with trust On the one hand, trust requires lack of information: if one were certain about future behaviour, we would no longer speak of trust. On the other hand, trust is based on information, in the attribution of motives and competencies to people, based on observed or reported behaviour"*
>
> *"Concerning the sources of trust, there are psychological causes and rational reasons. Psychological causes include emotions and may entail reflexes or automatic response.*

Rational reasons entail inference, on the basis of perceived behaviour, of someone's Trustworthiness"
"trust is taken here as a four-place predicate: the trustor (1) trusts a trustee (2) in one or more aspects of behaviour (3), under certain circumstances (4)."

(Nooteboom 2005)

If we summarise the sources, then we can see the following two main points:

- **Trust- why**: One trusts one another based on the information that you have about him. There three types of information sources on which one can trust someone. The first is direct information based on individual experiences with the transaction partner. The second is indirect, based on the experience of others. The third is based on the opinion of an expert about that transaction partner. So most trust models rely on either trust (based on direct information) or reputation (What others say about the transaction partner). There are however also mixtures of both.
- **Trust- in what:** A good question would be "In what does one trust one another?". According to the study of van der Leijden, we can trust someone in the fact that he or she has good intentions and/or that he has the right capabilities and competence.
- **Three central modes of trust production:** there is process-based trust in which it is tied to past or expected exchange such as in reputation or gift-exchange, there is characteristics based trust, where it is tied to a person, depending on characteristics and there is institutional based trust, Where trust is tied to formal societal structures, depending on individual or firm-specific attributes (e.g. certification as an accountant) or on intermediary mechanisms (e.g., use of escrow accounts).

(Leijden 2008)

In our study within the works of van der Leijden, we have also found a large set of trust models, which we can summarise by using (Sabater 2005). We will shortly note all of the concepts and give an overview on the next page (See for more details (Leijden 2008)) :

- **Conceptual model:** trust and reputation models can be characterised as either a cognitive model or a game-theoretical. In the cognitive approach, the mental states that lead to trust another agent or assign a reputation, as well as the mental consequences of the decision and the act of relying on another agent, are an essential part of the model. In the game-theoretical models, trust and reputation are not the result of a mental state of the agent in a cognitive sense but the result of a more pragmatic game with utility functions, and numerical aggregation of past interactions
- **Information source:** There are multiple types of information sources: direct experiences, witness information, sociological information (Based on relations, communities et cetera), prejudice (a mechanism of assigning properties (like for instance a reputation) to an individual, based on signs that identify the individual as member of a given Group).
- **Visibility types:** Trust and reputation of an individual can either be seen as a global property shared by all the observers or as a subjective property assessed particularly by each individual.
- **Model's granularity:** A single-context trust/reputation model is designed to associate a single trust/reputation value per partner without taking into account the context. A multi-context model has the mechanisms to deal with several contexts at a time maintaining different trust/reputation values associated to these contexts for a single partner. Nowadays, there are very few computational trust and reputation models that care about the multicontext nature of trust and reputation and even fewer that propose some kind of solution. This is because current models are focused on specific scenarios with much delimited tasks to be performed by the agents.
- **Agent behaviour assumptions:** there are three levels of model capacity to deal with agents showing different degrees of cheating behaviour. In level zero, cheating behaviour is not considered, in level one, an agent can hide or bias information, in level two the model has specific mechanisms to deal with liars.

- **Type of exchanged information:** the type of information expected from witnesses is either Boolean (within probabilistic models) or deals with continuous measures.
- **Trust/reputation reliability measure:** Sometimes, as important as the trust/reputation value itself is to know how reliable is that value and the relevance it deserves in the final decision making process. Some models incorporate mechanisms that provide this kind of information. The calculation process is often different per model.
- **Computational trust and reputation models**: the following models have been found:
  - *S. Marsh:* a computational model that only takes into account direct interaction. It has basic, general and situational trust. The trust values are used to help an agent to decide if it is worth it or not to cooperate with another agent. Besides trust, the decision mechanism takes into account the importance of the action to be performed, the risk associated to the situation and the perceived competence of the target agent. It also checks if an agent deflected in the past.
  - *Online reputation models:* reputation mechanisms such as eBay uses. With positive, negative or neutral evaluations of users over transaction partners (which are users as well).
  - *Sporas and Histos*: these are evolved versions of those evaluation systems. They alter the usage of the general reputation model by different modifications (which are out of the scope of this thesis)
  - *Schillo et al*.: a model based on a Boolean impression of the interaction between two agents: good or bad. Partners can be evaluated through a certain process and information about the results of that process can be exchanged with neighbours. One can bias information, yet not lie (since the neighbour might have actively participated in the process). One can only exchange information about a certain result of a process, not the average of all of the results of the different evaluation processes.
  - *Abdul-Rahman and Hailes*[78]: the trust model uses four degrees of belief to typify agent trustworthiness: *vt* (very trustworthy), *t* (trustworthy), *u* (untrustworthy) and *vu* (very untrustworthy). Agents can store the experiences and the information from witnesses based on their experiences in a tuple. An agent is evaluated to a certain value based on the highest frequency found in a tuple. If there are multiple maximum frequencies, then an uncertainty will be provided as well. Agents can adjust the information given by witnesses by their own experience with a certain agent.
  - *Esfandiary and Chandrasekharan*: there are two one-on=one trust mechanisms. One is based on observation (bayes analysis, see (Sabater 2005) for more details) and the other is based on interaction. There are two protocols of interaction: exploratory, to see if an agent is trustworthy and the query protocol to ask trustworthy agents for advice. Trustworthiness of an agent is determined in a networked fashion where it is calculated based on the values that other agents have. There is also a trust acquisition mechanism using institutions.
  - *Yu and Singh*: in this model, the agent stores the quality of the direct interactions with other agents (QoS) (see for a calculation of that quality (Sabater 2005)). There are two types of information that one can provide about a target agent: either QoS information (if the agent is known) or references to other agents, which might either return references themselves or QoS information. If the QoS information provided is not to far away in depth of the chain, then it will be used. If any direct QoS is available, then there will be no query for other info. The set of referral chains generated due to a query is a Trust Net similar to that by Schillo et al and in the Histos model.
  - *Sen and Sajja*: both direct interaction and observed interaction are used. The latter one is a little noisy and the first will give a true perception. So direct interactions are used to update the observation values if the impact is larger. The reputation ranges from 0 to 1, above 0.5 is trustworthy. Agents can query other agents about the trustworthiness and

[78] This is one of the two models that van der Leijden recommends in his study.

will either get a rounded answer (0 or 1). Agents can lie (only consistently) about others, so mechanisms are used to bypass them or filter them out partially.

- *Afras*: a computational model that works with fuzzy sets of trust. Once a new fuzzy set that shows the degree of satisfaction of the latest interaction with a given partner is calculated, the old reputation value and the new satisfaction value are aggregated using a weighted aggregation. The weights of this aggregation are calculated from a single value that they call *remembrance* or *memory*. The notion of reliability of the reputation value is modelled through the fuzzy sets themselves. A wide fuzzy set for a reputation value represents a high degree of uncertainty over that value while a narrow fuzzy set implies a reliable value. Recommendations from other agents are aggregated directly with the direct experiences. The weight given to each factor (old reputation value and new opinion) is dependent on the reputation that the recommender has.

	Conceptual model	Information sources	Visibility	Model's granularity	Agent behavior assumptions	Boolean exchanged information	Trust-Rep reliability measure	Model type
S. Marsh	GT	DI	S	CD	NA[a]	NA[a]	×	Trust
Online Rep models	GT	WI	G	NCD	0	×	×[b]	Rep
Sporas	GT	WI	G	NCD	0	×	√	Rep
Histos	GT	DI + WI[c]	S	NCD	0	×	×	Rep
Schillo et al.	GT	$_{DO}^{DI}$, WI	S	NCD	1	√	×	Trust
A.-Rahman and Hailes	GT	DI, WI[d]	S	CD	2	4 trust values	×	Trust Rep
Esfandiary and Chandrasekharan	GT	$_{DO}^{DI}$, WI, P	S	CD	0	×	×	Trust
Yu and Singh	GT	DI, WI	S	NCD	0	×	×	Trust Rep
Sen and Sajja	GT	$_{DO}^{DI}$, WI[e]	S	NCD	2[f]	√	×	Rep
AFRAS	GT	DI + WI[c]	S	NCD	2	×	√	Rep
Carter et al.	GT	WI[g]	G	NCD	0	×	×	Rep
Castelfranchi and Falcone	C	NA[h]	S	CD	NA[h]	×	NA[h]	Trust
ReGreT	GT	DI + WI + SI + P[c]	S	CD	2	×	√	Trust Rep

[a]There is no exchange of information between agents.
[b]The reliability is based on the number of ratings.
[c]The '+' symbol means the model combines the information sources to obtain a final trust/reputation value.
[d]Direct experiences are used to compare the point of view of these witnesses with the direct perception of the agent and then be able to adjust the information coming from them accordingly.
[e]Because the objective of this work was to study how agents use word-of-mouth reputations to select one of several partners, agents only use witness information to take decisions.
[f]Liars are assumed to lie consistently.
[g]Besides information coming from other users (WI) there is a central authority that monitors the agents behavior and uses that information to build reputation.
[h]In the description of the model it is not specified how the agents obtain the information to build their beliefs.

Figure 25: Overview of trust models. (Sabater 2005)

- *Carter et al*: it is built on the idea that *"the reputation of an agent is based on the degree of fulfilment of roles ascribed to it by the society"*. There is no universal calculation of reputation: each society has its own set of roles and contexts. The society calculates the overall reputation of a user as a weighted aggregation of the degree of fulfilment of certain rules (/role) such as: the rule of the social information provider (contribute new knowledge), the rule of the interactivity role (regularly use the system), content provider (provide the society with objects that reflect their own area of expertise), the rule of the

administrator feedback role (provide feedback information), the rule of the longevity role (holding your reputation high in other roles).

- *Castelfranchi and Falcone*: This model sees trust as a set of mental attitudes characterizing the "delegating" agent's mind (*x*) which prefers another agent (*y*) doing the action. *Y* is a cognitive agent, so X believes that *Y intends to do* the action and *y will persist* in this. (see (Sabater 2005) for more details)
- *ReGreT*[79]: is a modular trust and reputation system oriented to complex small/mid-size e-commerce environments. The system works with direct experiences, information from third party agents and social structures. It works with different knowledge bases such as, outcomes, information and sociogram databases in which all of the information is stored about its perception on the world. Trust is calculated based on different modules (the agent can use certainties and other decisive coefficients to decide which module should be used and what information should be trusted). The direct trust module deals with direct experiences and how these experiences can contribute to the trust on third party agents. The credibility module allows agents to measure the reliability of witnesses and their information. The reputation model uses witness reputation, neighbourhood reputation (information coming from social partners their relations) and system reputation (the reputation value is based on roles and general properties). An ontological structure provides the necessary information to combine reputation and trust values linked to simple aspects in order to calculate values associated to more complex attributes. For example, the reputation of being a good flying company summarises the reputation of having good planes, the reputation of never losing luggage and the reputation of serving good food. In turn, the reputation of having good planes is a summary of the reputation of having a good maintenance service and the reputation of frequently renewing the fleet. Each individual can have a different *ontological structure* to combine trust and reputation values and a different way to weigh the importance of these values when they are combined.

(Sabater 2005)

As we have seen in the footnotes: ReGreT and the model of Abdul-Rahman and Hailes are the ones that van der Leijden recommends to use in a environment based on the Jericho concepts. However, it is important to look at the other models as well, in order to understand what the literature can bring us in terms of trust models and if there is another model to consider when trying to find a solution for trust in the Collaboration Oriented Architecture Framework.

What is trust, based on the study of Andor Demarteau?

In (Demarteau 2008) trust is reviewed from the perspective of the Perceptual Control Theory (PCT). (See (Demarteau 2008) for more details around PCT itself). He studies a subset of the sources which van der Leijden (Leijden 2008) used and adds a few important aspects:

- **Risk Management:** He sees that Risk Management is an evident part of trust according to PCT. In order to trust someone, risk will have to be managed. Risk and trust are each other opposites. We trust that something positive will happen, yet we risk it going wrong at the same time. There is a trade-off point on which the trust-part is stronger than the risk of it going wrong. However, as this is outside the scope of this thesis, we will not explain this in detail.
- **Reputation:** reputation is considered a very important building block for building a trust relationship. In fact he later on defines trust as "an evaluated set of evidence gathered through observation and/or requested through recommendation", which is reputation.
- **Measurement of trust:** High-level, strong or lasting, as a longer duration of the period people refrain from harming the other person and/or having the wellbeing of that person in mind.

[79] This is the second model that van der Leijden recommends in his study.

This is one of the prerequisites for communication and maybe extended to other sorts of circumstances or simply to continue for a longer period of time or both

- **Link to concepts of cooperation and coordination:** He gives two examples of this by explaining about a bus ride and the mutual goals he has similar to those of the bus driver, which are low in the PCT hierarchy. The other example explains about a fire engine with two sets of steerable wheels and two drivers coordinating the movement of the truck. In both cases, some form of trust is involved. The cooperation itself must become an internal goal or reference so that both persons act to perceive that they are indeed maintaining the cooperative relationship.
- **Not entirely general:** People have varying motives (higher-level reasons) for wanting to be trusted, or to trust others. It is probably a mistake to generalise and try to find a certain characteristic that everyone has.

Recommendations for a trust broker framework from Demarteau

Demarteau has made the following recommendations in (Demarteau 2008):

- Reviewed systems:
  - *Unusable:* The following systems are unusable for Jericho trust management: Keynote and Policymaker: they are both encryption and certificate based and tend to have many problems in terms of trust and trust management.
  - *Could be usable:* The current reputation based systems could be interesting in terms of their current layered work (authentication and authorisation layer, accountability layer and anomaly detection layer), yet, even though they do not handle any real form of trust, they still could be used for trust management since reputation is an important part of trust.
  - *Interesting:* There are PKI systems developed with mediators, that look like trust brokers. These are very interesting, yet no real trust relationships have been defined. These systems will also give scalability problems when trust relationships will be implemented.
  - *The SECURE project:* the SECURE project tries to use a human notion of trust as basis for access decisions. With this goes the inexorable problem that trust is hard to define properly in the social sciences, let alone in a digital environment. Part of the solution is not only to look at trust but at the associated risk to transactions as well. Trust in the mean time is build up out of recommendations by other entities and observations done by the entity itself. These two sources of information combined form a dynamic opinion about another entity. In fact, the bigger idea behind the whole system is not really trust as such but one of the main, and definable, building blocks of trust: reputation. This is clear by the fact that all trust calculations in SECURE are based on evidence gathered mostly out of observations and where absent or not precise enough via recommendation as well. See for more details around the SECURE project (Demarteau 2008).
- **Requirements for a trust management system:** The following requirements have been given by Demarteau for a trust management system:
  - *Digital identity:* Identity is an important step in establishing a trust relationship. In order to trust someone, you will have to know who he is and that he is, who he says he is. Demarteau proposes a hierarchical system that can handle multiple methods of identification verification that can establish what you know, what you have and what you are. Furthermore, the digital identity systems should be user centric.
  - *Digital trust:* Demarteau sees that there is a need for observation of other entities (to see whether they are trustworthy) and recommendation about that entity (to see whether others see him as trustworthy as well). Both of the needs should be realised by controls, combined and compared on a higher level. He also sees the Circles of Trust framework from the Liberty Alliance (which is a digital legal framework for collaboration) as a part of the solution to gain more confidence in collaborating with a known entity.
  - *Trust Management system:* The following requirements have been made around the trust management system:

- Hierarchy of control: As the Perception Control Theory is based on a hierarchical approach, so should the trust management system. A two-layer system should do it for now.
- Levels of decision: there should be differential levels of decision: in the lower layers, no decision should be made, but results of those layers should be used in a higher layer to either deny or grant access and give trust.
- System Requirements: the interface between the two layers should be completely open, transparent and unchangeable. For every level in the hierarchy there should be a perceptual signal, a reference signal, an error signal and an action/behaviour control (see (Demarteau 2008) for more details).

(Demarteau 2008)

As one can see: all of this is based on the Perceptual Control Model. Most of the recommendations are based on this theory and therefore unusable if one wants to use another model or theory to define trust.

Trust and the position papers

The Jericho Forum has stated their vision on this matter in (Forum 2006f) . We can summarise it as follows:

- **Problem:** e-commerce transactions require a level of trust between participants. This relies primarily upon contracts and an enforcement mechanism (to punish and deter non-performance). A set of registration processes is necessary to register and verify each party's identity. However, the problem with registration processes is that they are hard to automate and, therefore, expensive. Splitting the cost by using federation, yet these mechanisms are oriented towards federating customer identity between members of a supply chain and, by agreement, between related supply chains. They aim to facilitate interactions between a customer and an organisation. One will need new mechanisms such as reputation, for sharing trust information and a common legal infrastructure, in the form of standardized contract templates, is required to facilitate de-perimeterised eCommerce. This situation directly links to JFC6, 7 and JFC8.
- **Trust definition:** trust can be defined according to the paper as:
  - *A verb* - A decision to rely upon someone's future performance of a contract; or
  - *A noun* - confidence that someone will meet a contract, based on his or her perceived capability, intentions and an accountability mechanism.
- **Trust and collaboration:** Trust is seen as vital for successful collaboration. The contract is central to the concept of trust. Accountability and identity are supporting to that contract. A party cooperates with another party because he believes in some combination of the following: the party is well disposed towards him, it is in the trusted party's best interests to comply, the trusted party has the necessary competence, skills and resources to comply an accountability mechanism exists that can force the trusted party to comply. In order to work together, organisations need to accept, and therefore understand, each other's contracts. Increasingly this will need to be done in an automated way. Businesses also need to be able to account for the contracts/authorisations they have agreed to (both as producers and consumers) in order to understand the obligations they are currently under.
- **Links to authorisation and authentication:** Authentication links an electronic agent to a real-world identity that forms the basis for an accountability mechanism; and authorization represents a degree of trust or competency that has been assigned to the identity. An authorization represents a contract, an agreed set of rules about how the holder and granter of the authorization will behave.
- **Reputation**: How does one party decide to trust another? It must decide whether the party is trustworthy or not, based on the proposed contract and a perception of the other party's past performance. Good performance in similar areas makes it probable that the other party will be trusted. A record of performance constitutes 'reputation' – good or bad. There are two mechanisms for "complete strangers": parties may share reputation information with

others they trust, allowing one party to take advantage of another's experience; or a party may choose to trust a stranger in a small way initially, based on global accountability mechanisms such as the law, and then escalate trust based on good performance.

- **Trust Architecture:** The recommended trust architecture is displayed as Figure 26. The concepts can be explained as follows:
  - *The Contract* is an agreement an organisation is considering entering into. This could be a business contract, or the allocation of a group membership in a directory
  - *The Prospect* is the other party in the contract. This could be a user applying for membership of a group
  - *Contract Approval* is the decision making process for whether or not to enter into the contract. It will use information in the reputation repository in making this decision
  - *Contract repository*: If the contract is signed, it will be entered into the Contract Repository so the organisation can monitor its assets and liabilities. The contract repository can be considered a part of the organisation's accounts. In many organisations, the contract repository is implemented as group memberships in an LDAP user directory.
  - *Behaviour monitoring and obligation monitoring*: As both parties execute the contract, a Behaviour Monitoring process ensures that the trusted party is complying to the contract; and Obligation Monitoring ensures that the organisation itself is complying. In the electronic world, this is implemented by access management, provisioning and user audit

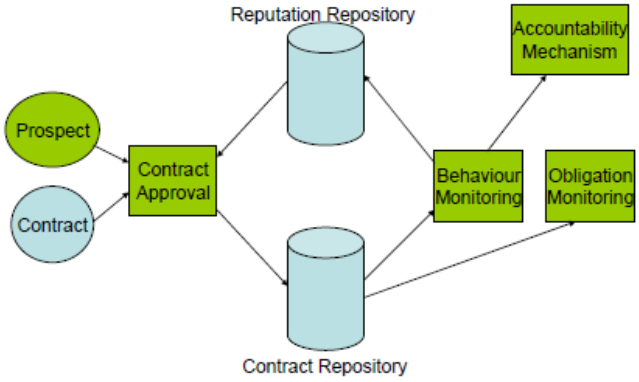


Figure 26: Trust architecture. (Forum 2006f)

plying to the

parties, their
cisions. It may

- one by several
  uch as direct
  recommend
  r mechanisms
  er concept as
  delegation of
  t will become
- dels and trust

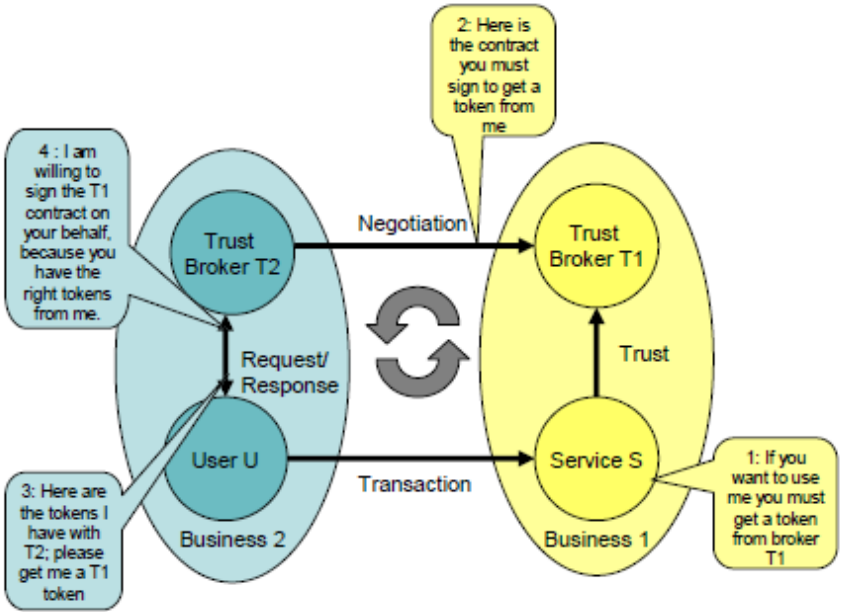


Figure 27: The trust broker as proposed in. (Forum 2006f)

COA V2.0 UPDATES:

The following relevant additions have been made in the COA 2.0 revision:

- **A global overview of what is necessary for trust management**, in: "Trust Management – A Brief Overview", which can be summarized as follows:
  - *There is a need for trust* between individuals as well as enterprises. The doomsday scenario is that the "Loss of Confidence" as in the current world of banking will occur on the web. One should prevent that from happening. Cost savings should be made as well.
  - *Necessary components:* one will need a number of capabilities, such as secure end2end communications, IAM- and trust management, commonly understood business impact levels, commonly agreed information sensitivity classifications, control stratification that is seen to work, enterprise relationship management. Most important is the need for open standards which will provide interoperability.
  - *Classification is a part of trust management:* the Jericho Forum considers data classification as a part of trust management. See section 2.6.6 for more details.
- **The new basic parts of trust management**, in "Position Paper – COA Framework", show a new outline of the trust management, which can be summarized as follows:
  - *Business impact levels*: there are 5 levels of impact proposed: catastrophic, material, major, minor, insignificant. See also the update at section 3.7.3.
  - *Information classification:* the information classification should be included, whereas they use the traffic light protocol.
  - *Impact sensitivity categorization*: there should be an impact sensitivity categorization of the information based on measures of it's confidentintiality, integrity, authenticity & availability, whereas the same 5 levels of the business impact levels should be used, with the additional level "none".
  - *Control Stratification:* a set of standardized information trust categories by trust level would be required. One could define a 6-level trust taxonomy for authenticity: Assured, affirmed, proven, confirmed, asserted and unknown. See the Position Paper for more details.
  - *Architecutre Segmentation Model:* a coherent architectural mdoel is required to map the Trust Management components into an effective operationally aligned structure.

The publications can be found at: https://www.opengroup.org/jericho/publications.htm

Open Groups Risk Taxonomy

In order to trust one another, additional Risk Management and a universal standard around Risk Management and the expression of risk will be necessary. There are currently a lot of activities on the field of Risk Management and a promising standard is (Fox 2008), it is related to many other documents that are in development as well. However, this standard is still in development and it is considered too premature to use it inside the scope of this thesis.

Trust broker services by Bruning

In (Bruning 2008b) Bruning has described a trust broker and the role of that broker. His work can, in relation to trust, be summarised as follows:

- **The basics of trust** can be summarised as follows: A trust relationship is build upon a logical and emotional act, measured by the ability, integrity and benevolence of somebody. Maintaining a trust relationship is done by continuously measuring these characteristics and evaluate if the relationship is still beneficial and or reliable.
- **Digital trust** will have to incorporate the same human needs: you want to know who the other person is, you want to know specific history details of the person to perform some kind

of Risk Management and you want some methods to manage the trust relationship so you can reduce the risk you are taking. The first one can be realised through authentication, the second cannot be performed without the help of somebody else with the appropriate access. The third can be done with digital contracts and digital signatures but these are not yet legally recognised.

- **Problems with digital trust:** there are several questions that rise on the field of digital identity: who is the person you want to build a relationship with? One can use authentication, yet, what if the account is hacked? How can ability, integrity and benevolence be measured in the digital realm? You will have to be authorised to private information. Yet, we will remain uncertain if we can always trust the source that provides that information. How can one keep on evaluating the trustee? How can we be certain if you trust someone without breaking his privacy? How can you control and manage the trust relationship, without legislative support?
- **Recommendations for digital trust:** The following recommendations have been made:
  - *Check information:* after you know someone's identity you must be able to check this information, how else can you otherwise be certain if somebody is who he says that he is.
  - *Amount of information:* depending on the kind of trust relationship, you want to establish (more or less trustworthy), different amounts of information and level of detail will be necessary.
  - *Re-evaluation:* You have to be able to continuously check the information about the identity where you have a trust relationship with, for as long as the relation exists.
  - *Law:* legislation should be developed on the field of digital identity and properly enforced.
- **The trust broker** has as a main purpose, to act in trust. It will establish and manage trust between parties. A neutral third party offers services for financial compensation. The trust broker will have to determine if every party can be trusted and if it is still the same party when the agreement was concluded. It will use the circle of trust. It will furthermore:
  - *Manage sensitive information:* The trust broker will manage the sensitive information of the organisations that are connected to it.
  - *Be responsible*: it will take full legal and financial responsibilities for any loss or compromise of the information.
  - *Provide control tools*: it will provide better control tools to companies in order to give better insight and decisiveness within the matters of securing and storing personal sensitive information.
- **Trusting a trust broker:** a trust broker can be trusted because of the following reasons. First, a company will not become all-powerful by knowing everything about you and with whom you deal. This is because of its architecture (See also (Bruning 2008b) for more information). Second, such a company will be paid to do a good job, which means handling personal sensitive information with care and giving their utmost to protect it. Third, if companies are compliant with laws, people can presume that the company will take great care of their personal sensitive information.
- **Trusting an entity:** A trust broker can see an individual, a company and a device as an entity that can be (dis-)trusted.
- **Functions of a trust broker:** the trust broker will have the following functions:
  - *Identification:* the trust broker will have to know the identity; it can do so by delegating the identification services to existing network functions or it can create an extra service. This is done by authentication.[80]
  - *Retrieval and checking of additional information*: there are two types of information that need to be checked: self –asserted and given by an authority. This can be done by cooperation with other services, which provide the necessary claims, credentials or attributes about an entity.

[80] See section 2.6.7for more details.

- *Verifying and updating information about an identity*: verifying should -and updating could- be done by the trust broker. Information with high risk should be verified by an authority and with a low risk by its own sources.
- *Legislative control*: legal support can be implemented by means of contracts. The contracts can be made digitally but have to be signed on paper.

- **Roles of the trust broker:** the trust broker must become the central gateway for users to access data or services that are provided by a company. It should provide all processes to create a good and secure connection between the user and the company. This complies to JFC6 and 7, because trust is applicable to all entities, the trustworthiness behaviour for establishing transactions is ensured by contracts, mutual trust assurance levels can be determined with the combination of a legal framework and the different trust broker models. In order to be in line with JFC10, segregation of duties will have to be implemented. This will make it more difficult to perform deliberated fraud, because in order to complete a transaction the involvement of two or more parties is needed. This will be implemented by isolating different processes in the trust broker such as authorisation and authentication. Since it is designed for external transactions, it can be added to the corporate network that is already optimised for internal transactions. It will have to be added in such a way that it can cooperate but cannot control every process. Every process has to be capable of sending the request to the external services. It has to comply to JFC2 as well, so it needs to be scalable. Reputational and behavioural services and servers can be separated from the trust broker, because they are dedicated systems that are constantly monitoring and collecting potentially sensitive data about the entities. Furthermore, multiple ways of authentication can be established, within/by the corporate network or by means of another external server. Finally, the legal control is very important and should be done by the trust broker by means of digital signatures, stamps and other measures.
- **Suggestion of a trust broker framework:** a trust broker framework is suggested, which will work like the ws-security protocol. It will be build upon meta-data system were attributes can be simply exchanged between different kinds of protocols. It should be open and interoperable with other systems and communicate by means of XML. It should comply to JFC4 in that matter. A modular system will have multiple qualities and advantages (see (Bruning 2008b). There will be security vulnerabilities, which can be encountered by (new) inherently secure protocols and standards.
- **Recommendations of legal frameworks:** The legal frameworks of project liberty (circle of trust) are recommended to be used. The trust broker can come to its full potential within these frameworks. See (Bruning 2008b) for more details. The setups can be combined with several trust broker model designs. Both are out of the scope of this thesis. See (Bruning 2008b) for more details.
- **Trust broker models:** There are different trust broker models:
  - *Central trust broker:* a star network topology with the trust broker in the middle. The trust broker will handle all the activities around federation, global security, policies, keeping of records around the reputation and the monitoring of obligation between parties. It is relatively simple, can do all the intermediary functions and allows dynamic structures in the model, however, it can become a single point of failure and is more likely to be subject of fraud attacks.
  - *Server/client side Trust broker:* a server trust broker and a set of client trust brokers take the role of the complete trust broker. The clients will do the record keeping, enforcement of global security measures and enforcement on its own hosts. The server will do the federation, keeping copies of the clients, keep records of reputation et cetera. It can create a push& pull model, where more information sources are accessible and available, creating information that is more trustworthy. One will however have to redesign his corporate network to incorporate a trust broker.
  - *Peer-to-peer trust broker:* There is a set of peer trust brokers that are connected and will perform the same tasks. Some of the functions should be delivered by the community such as managing identities and keeping copies of all of the obligations. There is no

obvious server, reducing risk of single point of failure and the p2p structure and a fully connected network creates opportunities to perform additional functionalities, however it will be difficult to manage.

- **Trust broker functionalities**: the following technology-trust broker functionality mapping has been found:[81]

Trust broker functionalities	Technology	Note
Identification	OpenID	Creates a decentralized Identity Management system, that will l et you use multiple identities.
	MS Cardspace	Creates a simple front-end system to choose your identity. Made possible by the identity meta-system of K. Cameron.
	Higgins	Open source project to create a true platform- and protocol independent framework for a secure identity infrastructure
	XRI	XRI is a true RESTful approach. XRI is a universal identifier for all entities across multiple domains or directories.
Reputation modelling	Jyte eBay and Experian	A combination of these technologies with additional functions for analyzing information collected by the accounting process
Behaviour monitoring	XDI / Link-contracts	As a sub-part of XDI, link-contracts make it possible to monitor a digital contract by giving the owner active control.
	WS-agreement	This protocol enables to create, specify and manage an agreement with other parties.
	WS-policy	Makes it possible for web-services to advertise a policy through XML.
	WS-trust	Makes it possible to create, exchange and validate different security tokens by using the security mechanism from WSS.
	WS-federation	Specifies mechanisms to allow different security domains to federate.
	ID-WSF	ID-WSF is a framework based on open standards and specifies how to make a secure, multi-vendor federated network.
	SAML 2.0	SAML promotes interoperability between disparate security systems, providing framework for secure e-business transactions across company boundaries

Table 4: Mapping of trust broker functionalities and current existing technologies, part 1.

- **Trust brokers and identity**: Again, the user centric model is recommended for use with a trust broker framework.
- **Feasibility of the trust broker framework**: new technology needs to be developed, yet none of it is will be shocking. Even though most of the problems will come from the technological possibilities. In terms of business process, many changes will occur. Another aspect will be the costs of using the circle of trust and a trust broker framework. The trust broker framework will provide many benefits, which would justify the costs.

(Bruning 2008b)

Trust broker framework by Bruning

Bruning continued his work in (Bruning 2008a). He set up a set of requirements in this study, which can be summarised as follows:

- **Legacy and modern systems:** The trust broker framework should be capable of working with legacy and modern systems.

[81] See the sections around Identity Management and the sections around SOA for more details around most of the technologies. See Furthermore, (Bruning 2008a) for more details as well.

- **Information and contracts:** all required information for determination of a reputation of an entity should be stored. Contracts must be enforceable and compensation mechanisms should be included.
- **Functional requirements:** the following functional requirements have been found:
  - *Broker of requests:* the trust broker is able to handle all kinds of requests and transform these in events and triggers.
  - *Generates Trust Context Reports/Profiles for entities:* the reports/profiles are made based on the retrieved values for different security attributes assigned to the data, user, end-point et cetera based on the last logs.
  - *Is able to perform a trustworthiness check:* this is done by an identity check to create a contract.
  - *Generates contracts:* if the circumstances demand for additional security controls, a contract is generated based on the business needs and requirements of the company or companies.
  - *Discovery service:* is able to select the best fitting service for the job. The discovery service is able to select a service based on the information it has about the entity. This information is compared to the obligations and policy it has to comply to.
- **Non functional requirements:** The following non-functional requirements have been found:
  - *Scalability:* the trust broker framework must be able to scale in order to support a wide variety of services.
  - *Flexibility:* the trust broker framework must be able to adapt quickly to new situations e.g. quickly support a new service within the network (Mashups).
  - *Auditability*: all operations and results of the services executed within the trust broker framework must be audited.
  - *Transparent*: the trust broker framework must be open and transparent so any one authorised can easily traces processes and determine bottlenecks or security flaws.
  - *Performance*: the trust broker framework must be able to quickly retrieve and analyse essential information.
  - *Governance and compliance*: the trust broker framework must have monitoring and control tools in order to easily implement low-level control objects that arise from the different quality control certifications.
- **Security requirements:** the following security requirements have been found:
  - *Segregation of duties:* in order to ensure that the implementation and execution of the modules or processes is not violated or abused, the rights and privileges regarding their execution must be separated.
  - *Trusted sources:* each individual module/process must use verified and trustworthy sources.
  - *Secure communications:* communications and interactions between the other modules/processes identified in Jericho Security Architecture, or between Trust broker services and the identified modules/processes must be adequately secured, since large quantities of sensitive information will be dealt with.

(Bruning 2008a)

He also gave further elaboration on the field of the Trust broker framework: how it would look like and how it would function. However, that is outside the scope of this thesis for now. See (Bruning 2008a) for more details.

Concluding:
In this section, we have studied different sources around trust and trust management. We can conclude the following:

- **Trust** can be defined in multiple ways. If we take some of the definitions together, then we can say that trust is “a particular level of the subjective probability with which one assess that another entity or a group of entities will perform a particular action before one can monitor

the action. This action will be beneficial or at least not detrimental to the one that trusts the entity or entities".

- **Sources of trust**: there are several sources of trust:
  - *Rational:* Within rationality, we can find many sources: inference, on the base of perceived behaviour (experiences from the past) or the reputation of someone provided by the subject or third parties such as other persons or experts, of someone's trustworthiness. Furthermore, trust can be rationally raised by the usage of contracts and the enforcement of the contracts.
  - *Psychological:* these include emotions and many entail reflexes or automatic response.
- **Sources of trust need to be re-evaluated** in order to continue the trust relationship. This means that a trust relationship will have to be maintained by continuously measuring the defined sources and evaluate if the relationship is still beneficial and or reliable.
- **Central modes of trust:** The following modes of trust have been found: process based, characteristics based and institutional based trust.
- **Trust and collaboration:** trust is necessary in order to establish and maintain a collaborative relationship. There are sources that see the contract in association with identity and accountability between the parties as the central to the concept of trust.
- **Digital trust**: digital trust will have to incorporate the same human needs as real life trust. However, there are several problems on the field of digital trust such as identity, integrity measurement and the continuous evaluation of the trustee.
- **Recommendations for digital trust**: we found a set of recommendations on the field of digital trust such as the need for information checking and verification based on the type of trust relationship, the need for re-evaluation of that information and the enforcement of law on the matters surrounding trust and trust management.
- **Trust models:** various trust models have been found and summarised in this study. The most important ones that could be used in order to keep a certain automated bookkeeping would be the model of Abdul-Rahman and Hailes or the ReGreT model. However, looking at all of the other models, it seems clear that the final answer to trust modelling has not been given, yet many ideas, views and concepts should be taken in mind.
- **The need for trust and better trust management measures:** there is currently a high need for trust in many collaboration processes such as e-commerce. This is established by a set of processes, which are costly. Current federation mechanisms will not suffice and other measures such as a common legal infrastructure, reputation and other contract-based elements are necessary.
- **Trust management and architecture:** trust will have to be managed by a management system based on a trust model and a trust architecture. The Jericho Forum has provided a certain trust architecture, which is reputation based, alongside with other mechanisms such as accountability mechanisms, behaviour and obligation monitoring.
- **Trust management and the trust broker:** A very important element in such a trust management system is a trust broker, various analysis and recommendations have been made:
  - *The role of the trust broker:* is to establish and manage trust between parties, this will be done by several functions such as the management of sensitive information, taking responsibilities and providence of control tools to collaborating and related organisations.
  - *Functions of a trust broker:* the following functions have been defined: identification, retrieval and checking of additional information, Verifying and updating information about an identity and legislative control.
  - *Two-way trust:* there are various reasons why an entity or organisation should trust a trust broker. Furthermore, the trust broker itself can see a company, an identity or a device as an entity that can be (dis-)trusted. This means that there will be two-way trust.
  - *Trust broker models:* there are several models such as the central trust broker, server/client side trust broker and the peer-to-peer trust broker.
- **Trust broker and trust broker framework:** A recommendation has been found to create a trust broker framework that would work like a WS protocol. Legal frameworks as the Circle of

Trust from project Liberty could be included in the framework. Multiple issues and requirements have been found:
  - *Technology development:* there are multiple technologies for the current trust broker functionalities. Yet more technology will have to be developed in order to create the trust broker framework.
  - *Feasibility:* as stated, more technological developments will have to be done. These will be costly. Furthermore, many business processes will have to change, which will make it even more challenging. However, the cost can be justified, taking the benefits of trusting each other into account.
  - *Requirements:* the following requirements have been found:
    - Both legacy and modern systems should be supported by the trust broker framework.
    - Functional requirements such as being a broker of requests, the ability of performing trustworthiness checks, the ability of contract generation and service discovery.
    - Non-functional requirements such as scalability, flexibility, auditability, transparency, performance, governance and compliance.
    - Security requirements such as segregation of duties, secure communications and trusted sources.
- **Links to authorisation and authentication:** trust is linked to authorisation (a degree of trust or competency that has been assigned to the identity) and authentication (as a basis for the accountability).
- **Trust and PCT:** when one looks from the PCT perspective, then aspects such as Risk Management and reputation will become more important. Trust is seen as stronger as the period wherein two or more entities refrain from harming each other. There are also links to other concepts such as cooperation and coordination: both are linked to trust. Trust itself is not entirely general when one looks at it from the perspective of PCT: there are motives (higher-level reasons in PCT) for someone to want to be trusted.
- **Trust management systems and PCT:** different systems have been examined: the current certification based systems will not suffice and the current PKI and reputation-based systems are interesting for various reasons. The SECURE project has been reviewed by Demarteau as very interesting and as an example and base for the trust management system he wants to see.
- **Requirements for a trust management system from the PCT perspective:** Requirements have been made such as: the need for a digital identity, a hierarchical system that can handle multiple methods of identification, the need for controls at a higher level for the management of observation of and recommendations on entities, a set of system requirements and the recommendation for a layered system with several levels of decision.
- **Interesting concept for further research:** the Risk Taxonomy project is very interesting and should be used in further research projects for trust- and Risk Management.

2.6.9. End-point security

Introduction:
In this section, we will investigate some of the basics around End-point security. End-point security is important, since it is about raising the level of inherent trust in computing devices to a point where all the devices involved in a transaction meet the criteria of trust for that transaction. Therefore, it is directly linked to JFC1 and JFC6. (Arnold 2008)
In this section we will take a look at End-point security from the perspective of the following sources: the position papers, "Authorization and End-point Security", by Teheux (Teheux 2008), the Jericho Security Architecture by Stan (Stan 2008b) and the Jericho Forum Commandments. We will end the section with a little concluding summary.

End-point Security in the Position papers – definitions and explanation:

The Jericho Forum has published a Position paper around this subject (Arnold 2008). As seen in the introduction, they state the following about End-point Security:

> *"End-point security is about raising the level of inherent trust in computing devices to a point where all the devices involved in a transaction meet the criteria of trust for that transaction. (JFC#1 & 7) The trust level needs to vary in accordance with a range of factors, including risk, transactional value, location and time."*
>
> (Arnold 2008)

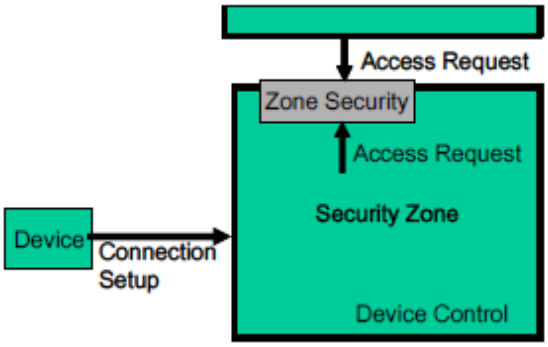


Figure 28: Current situation. (Arnold 2008)

In order to transact with another party (or parties), there needs to be a level of mutual trust between them (JFC#7), commensurate with the transaction that is to take place (JFC#6). This will allow valuable transaction to take place more safely.
The flexibility of having devices from multiple organisations or users being able to have their trust level validated upon trying to transact with your applications (as opposed to validation when they try to connect to your network zone), enables more flexible and secure ways of working. (Arnold 2008)
End-points can be network devices (the hubs, switches and routers that transfer network traffic around the network), access devices (the workstations, laptops, PDAs and mobile devices through which users access a network) and servers. An end-point's security posture is any security attribute for the end-point that a remote party may wish to rely upon such as: that is appropriately hardened, that the virus checker or IPS is up to date and/or that it is based on appropriately certified software. (Arnold 2008)

A small addition:
Based on interviews, we can say that one of the most important things to add on this position paper is the aspect of "system hardening": one should automatically apply all the latest software updates to the system(s operating system, firewall, virus scanner, security agent, etc.) and use the proper protection mechanisms at the end-point device.

End-point Security in the Position papers –The current situation: issues and recommendations:
The current situation is as follows: the end-point security focuses on managing end-points and network security boundaries or "zones" together. A security zone is a group of devices together under a common security contract. (See Figure 28 for more details)
There are different systems, however the basic concept is still that a device will connect to a security zone, and that the end-point security system will check the device if it is in line with the end-point's security posture. If so, then it can connect to the zone, and if necessary communicate through with other devices in the zone or with other end-points in other zones.
The paper describes the following issues and recommendations:

- Issues:
  - *Zone security device is bottleneck:* Generally, traffic can only leave the zone through a zone security device, this results in a single point of failure, a susceptible DOS attack point and, depending on where the control point is placed, a not efficiently working Internet routing.
  - *Lack of support for protocols:* many end-points do not support 802.1x, or need an agent installed to measure security posture and thus need special management.
  - *Interoperability between software (agents):* agent based software is required, use between different organisations is difficult as the agents may not interoperate. Different

agents are likely to clash, and "on-demand" installation of agents is unlikely to work where the end-point is locked-down.
  - *Trust is one way:* End-point security is generally limited to validating clients trying to connect into "your" environment, with the trust being one-way; the client not always being able to form an opinion regarding "you" even though you have established the means to gain an opinion about the client. Such one-way trust leads to attacks such as phishing.
- Recommendations:
  - *Multiple end-points need to be capable of registering at multiple zones:* End-points from different organisations should be capable of being registered in multiple organisational zones.
  - *Identity Management for end-points*: many of the Identity Management services currently being developed for users (registration, federation, single sign-on) are also required for end-points. User agents need to be able to access not just user credentials and tokens, but end-point credentials and posture checking agents as well. Similarly, access management services must make access decisions based on both user and end-point attributes.
  - *Two way trust:* The current browser "sandbox" concept needs to be expanded from one-way trust to support two-way trusts, thus allowing a device to make a secure connection and interact; with each party able to validate that the other is appropriately isolated (JFC#6 and 7).
  - *Validation of trust:* For systems that interact using just inherently secure protocols, then both systems must be capable of validating the trust, via a standard secure protocol, either directly, or more likely through a trust broker.
  - *Open standards*: There are open standards necessary for all of the above recommendations, such as the TNC specification – IF-TNCCS-SOH and the IETF Network Endpoint Assessment – RFC 5209.

  (Arnold 2008)

End-point security according to Teheux: definitions and current situation
Leon Teheux, a fellow researcher at the Security and Innovation Research Centre did a research project around End-point Security and authorisation (Teheux 2008). We will use this project as the second cornerstone for our research around End-point security.
He defines the End-point security process as being the

> *"responsible process for providing the means to establish inherent trust levels between end-points, with the intent to create a situation where all the devices involved in a transaction meet the criteria of trust for that transaction."*
>
> (Teheux 2008)

There are many End-point security-, or Network Access Control solutions, yet most of these solutions were not designed to interoperate with other solutions and the lack the ability to verify all network devices.

The requirements for End-point security by Teheux:
Teheux defined the following logical requirements in (Teheux 2008):

> *"**Trust***
> *Endpoint security must determine trust levels for all relevant devices*
> *Endpoint security must span all tiers of the architecture*
> *Endpoint security must be able to support mutual trust*
> ***Security***
> *Endpoint security must be able to protect the device it is operating on.*
> ***Scope***
> *Endpoint security must be able to operate with a global scope*
> *Endpoint security must be scalable*

Manageability
Endpoint security should be simple to manage.
Endpoint security should be managed as close to the device as possible."

(Teheux 2008)

See for more details around the process of definition (Teheux 2008). He also defined a set of technical requirements:

*"**Operation***
Agents must be able to operate on all relevant devices.
Agents must be able to communicate with any other Agent.
Endpoint status may be delivered in claims.
No single point of failure must exist.
Protocols
Endpoint security should use secure protocols.
Protocol used must be transparent and be able to be replaced.
Management
Segregation of duties should be implemented.
It must be possible to apply rights to external devices accessing endpoints under your control.
Standards
All protocols should adhere to open standards."

(Teheux 2008)

Recommendations by Teheux:
The following logical recommendations have been made:

- **Process interactions:** the End-point security process should be communicating with the user authorisation process and the processes around encryption, so that the level of trust can be defined (and communicated with the trust broker).
- **Logical architecture:** In order to make sure that all of the end-point security system scans interoperate, a logical universal end-point security architecture should be made. The following recommendations have been made for a logical architecture:
  - Peer to Peer:
    - Description: in a secure peer-to-peer architecture, each of the end-points mutually exchanges verification information. All peers act as equals, there is no central server or router.
    - Advantages:
      - All nodes on the network are part of the system, eliminating any single points of failure in the system, increasing availability.
      - All nodes provide their own bandwidth and resources, thus increasing accessibility
      - As no additional hardware is required, a lower cost.
    - Disadvantages:
      - A decentralized architecture inherently means lack of administrative control of End-point rights and rules
      - Additional administrative effort may be required to incorporate systems within the Accounting process.
      - All end-points must communicate using the same protocol.
      - A decrease in user privacy, as all end-points can request information from any other end-point.

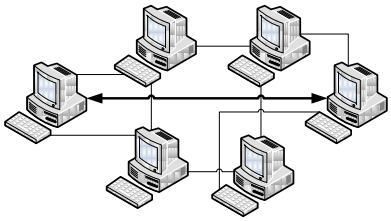


Figure 29: Peer to Peer.

  - Hybrid Peer to Peer:
    - Description: is like peer to peer, only now has a central server that keeps information on peers

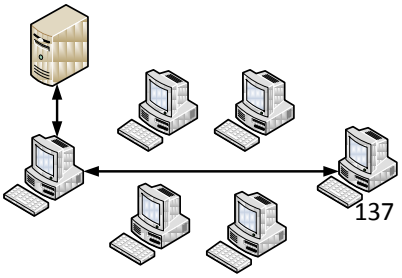


Figure 30: Hybrid Peer to Peer.

and responds to request of that information. The peers are responsible for hosting available information, for letting the central server know what information they want to share, and for making its shareable information available to peers that request it. The requester for information (or initiator) will contact the server to obtain a token with which it can access another peer to request information. The server checks if the client is allowed to ask for this kind of information and if the client from which it want information wants to share the requested information.

- Advantages: it negates the administrative issues and the decrease in user privacy from the pure peer-to-peer model.
- Disadvantages: it needs a server that could be a single point of failure.

- Trust broker:
  - Description: Multiple servers communicate with each other in order to provide End-point Security status for end-points. The trust brokers are responsible for determining and communicating the status of end-points under their control (See for more information section 2.6.8 around trust brokers). An end-point sends a request for verification to its local trust broker. This trust broker then queries the trust broker in charge of the remote end-point, which in turn verifies the end-point status and replies with the verification status.

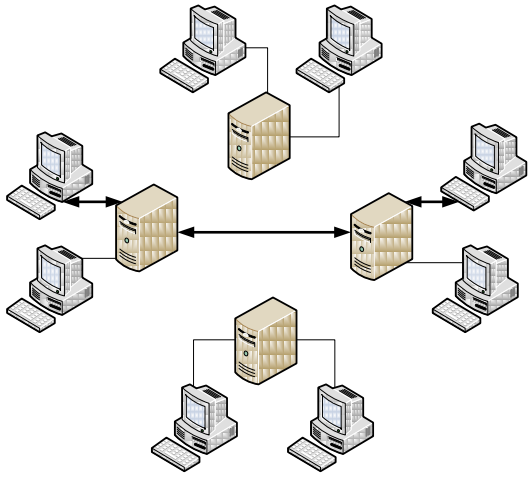


Figure 31: Trust broker.

  - Advantages:
    - The centralized architecture means administrative control remains within the organisation end-points belong to.
    - The ability to customize applications, as the organisation is completely in charge of its local End-point Security solution.
    - A centralized architecture decreases the amount of network routing needed, increasing responsiveness of applications.
  - Disadvantages:
    - A decrease in availability as the Trust broker responsible for intercompany communications may introduce a single point of failure;
    - An increase of nodes or usage of the network may slow down the network, decreasing accessibility.
    - A new infrastructure supporting the trust broker is required.
  - *Recommended logical architecture:* based on the logical requirements (availability, scalability and manageability), the Trust broker architecture is recommended. Even though it scores as the lowest on availability, it scores very high on the field of scalability and manageability.

- **Open standard:** there should be an open standard for establishing End-point security.

Based on a long selection, he also recommended a technical solution for the End-point Security in terms of a setup for a prototype: a network access control system that could meet the requirements for End-point Security, which should be connected to a trust broker. He recommended the Cisco NAC Appliance and as an alternative the Mirage Networks' End-point Control: both would be capable of being modified in such a way that they could connect with a prototype of a trust broker. (Teheux 2008)

End-point Security and the Jericho Security Architecture

Stan has given the following requirements and required capabilities for end-point security systems in (Stan 2008b):

- **Capabilities:** The following required capabilities have been given:
  - *In relation to data classification:* The end-point should provide and request reports about the data classification method and the process.
  - *In relation to encryption methodology:* It should provide and request encryption related data (e.g. algorithms, keys, protocols, et cetera)
  - *In relation to the trust broker framework:* It should request reports about the compliance of entities with contracts and security policies.
  - *In relation to authentication and authorisation mechanisms:* It should be capable of requesting and providing end-point status information. It should be capable of providing an authorization or a rejection response based on that information.
  - *In relation to accountability:* The security state should be logged and reports of the security status scans should be saved.
- **Requirements:** The following requirements have been stated:
  - Functional requirements:
    - The chosen solution should be able to protect and assess the security status all the devices on the network
    - The chosen solution should retrieve current information status about all the devices on the network
    - Interoperability. The End-point security solution should be able to provide information status for all sorts of devices on a network that operate in different environments
    - Scalability. The End-point security solution must be able to operate on a global scale, on any network device
  - Non-functional requirements:
    - Auditability. All End-point security checks and reports must be audited.
    - Availability. End-point security solution should provide high level of availability.
    - Cost effective. The End-point security solution should be a cost effective solution
    - Flexibility. The End-point security solution should deliver flexibility and should be easy to manage
  - Security requirements:
    - Audit logs recording all events and processes should be produced and kept
    - Security patch management policy
    - Segregation of duties/privileges for preventing the modification of the settings and the status data of the End-point security solution
    - Cryptographic controls (algorithms, primitives) for encrypting the status data that is generated and transmitted by the end-point security solution. The algorithms and primitives chosen should have not been yet broken and should provide strong protection (confidentiality, integrity).
    - The information status delivered by the End-point security solution should be expressed/reported in terms of security attributes.
    - A trust context is created and validated based on the security attributes delivered by the End-point security solution
    - Segregation of duties
- Analysis and recommendations:
  - *"As is" now:* The solutions of today are non-interoperable and not comprehensive.
  - "To be" target:
    - The security of all devices on the network should be verified and validated upon requests.
    - Agents on each device that monitor end maintain the device's security.
    - Automatic patches of the security mechanisms that protect the devices.
    - Automated corrective actions that enforce the security patch management policy.
  - Results Jericho research:

        - The security status of all devices on the network should be verified and validated upon requests
        - Agents on each device that monitor end maintain the device's security
        - Automatic patches of the security mechanisms that protect the devices
        - Automated corrective actions that enforce the security patch
    - Possible technologies: Lancope and NAC.

(Stan 2008b)

End-point Security and the Jericho Forum Commandments

If we look at the End-point security from the Forum Commandment perspective, then we cannot find any additional means to what has already been stated by the authors above.

Concluding:

Looking back at this section, we can conclude the following:

- **End-point security** is about raising the level of inherent trust in computing devices to a point where all the devices involved in a transaction meet the criteria of trust for that transaction. That level of trust is based on the status of the firewall, encryption standards and other aspects of the system hardening.
- **End-points** can be network devices, access devices and servers.
- **End-point security postures** are security attributes for the end-point that a remote party may wish to rely upon. Desired outcomes may be that it is appropriately hardened, that the virus checker or IPS is up to date and/or that it is based on appropriately certified software.
- **Current issues:** There are multiple issues in the current situation such as:
    - the zoned security device that is often the bottleneck,
    - the lack of support of security protocols such as 802.1x,
    - low interoperability between software (agents) on the end-points and
    - The zones and trust establishes are often one way: a client ties to connect to an environment, without the client completely checked.
- **Requirements:** The following requirements have been formulated:
    - *Trust:* end-point security must determine trust levels for all relevant devices over the complete architecture and it must be determined mutually.
    - *Security:* End-point security must be able to protect the devices it is operating on.
    - *Scope:* It should work with a global scope and it should be scalable
    - *Manageability:* It should be simple and managed as close to the device as possible.
    - *Information requests:* end-point devices should be able to retrieve current information status about all the devices in the network.
    - *Operation:* Agents must be able to operate on all relevant devices in the network and should be capable of intercommunication, there should not be a single point of failure.
    - *Protocols:* The protocols should be secure and transparent, capable of replacement.
    - *Management:* segregation of duties and distributed rights application should be implemented.
    - *Standards*: al protocols and standards should be open.
    - *Interoperability*: solutions for End-point security should be capable of interoperating.
    - *Other requirements:* The solution needs to be scalable, auditable, always available, cost effective and flexible.
    - *Recommendations as security requirements:* recommendations around policies, logs, cryptographic controls, trust, segregation of duties and (status) information exchange have also been found as requirements.
- **Recommendations:** The following recommendations have been made:
    - *Multiple end-points/multiple zones:* End-points from different organisations should be capable of being registered in multiple organisational zones.
    - *Identity Management:* Identity Management services should also be included in solutions for End-points.
    - *Two-way trust:* Current one-way trust solutions should be upgraded to two-way solutions.

- *Logical architecture:* One should use the trust-broker architecture for establishing and verifying end-point security.
- *Integration in other processes:* The end-point security should be integrated in the authorisation process. It should further more be capable of the following:
  - Report ad request information about the data classification method and the process.
  - Report ad request information encryption related data.
  - Report and request information compliancy and security status information.
- *An open and inherently secure protocol/standard set:* End-point security should be realised by the use of open and inherently secure protocols and standards.
- *Validation of trust*: systems should be capable of validating the trust level on the end-points and between the end-points.
- *Agents*: The security of all devices on the network should be verified and validated upon requests by agents, which will have to be installed on each device in order to maintain the security status. This will be done by corrective actions, security patches and other security mechanisms and services.
- *Automation:* There should be an automated (policy/) system for updating and maintaining the device status.
- *Currently recommended technology solution:* One could use either a Cisco NAC or the solution from Lancope. Yet both of them have to be altered to be capable of communicating with a trust broker solution.

2.6.10. IT-Audit

Introduction:
This section will focus on the field of IT-auditing in perspective to de-perimeterisation. IT-audits are indirectly related to JFC6 and JFC7, since they are focussed on reducing risks on the field of IT and indirectly on the field of business risks.
The first notice around this subject was made in (Forum 2007d) and in the relating paper (David Lacey 2006) and later on, they added the promised position paper (Henry S. Teng 2008).[82] The position papers deal with the issues and recommendations around IT-auditing in a de-perimeterised environment.
We will first look at what IT-audit is and then see the Jericho Forum has said in their Position papers around this subject. From there on, we will end with a little summary of this section.

IT-audit:
In order to understand the influence of de-perimeterisation on IT-audit, one will first have to know what IT-audit really is. The first published position paper from the Jericho Forum defines IT-auditing as follows:

> *"IT audit is about the formal verification and validation of the quality and effectiveness of IT controls to support the overall business control objectives."*
>
> (Forum 2007d)

There are multiple kinds of audits on this field. One could use the following subset of the audit types as defined in (David L. Cannon 2006):

- **Operational audit:** Verifies effectiveness and efficiency of operational practices. Operational audits are used frequently in service and process environments, including IT service providers. An operational audit is detailed in Statement of Accounting Standard 70 (SAS-70)

[82] We should make an important remark on this subject. The position papers that are used in this section are mostly still in draft or concept version while writing this thesis.

- **Compliance audit:** Verifies implementation of and adherence to a standard or regulation. This could include ISO standards and all government regulations. A compliance audit usually includes tests of presence of a control.
- **Administrative audit:** Verifies whether appropriate policies and procedures exist and have been implemented as intended. This type of audit usually tests for the presence of required documentation.
- **Information Security audit:** Verifies systems for certification and/or accreditation. Certification usually involves system testing against a reference standard, whereas accreditation represents management's level of acceptance.

IT-audit is also a basic part of IT Governance. We have seen many models, frameworks and principles around IT Governance in section 2.4.2. Most of the models and frameworks hold controls that can be checked. IT auditing is focussed on both making sure that the control is installed and that it is functioning. (David L. Cannon 2006)
All the details about internal or external auditors, the auditee, the different frameworks that can be used, are out of the scope of this section.

Questions and issues around IT-audit and de-perimeterisation:
In (Forum 2007d) the question is raised whether the tactical/operational aspects of IT audit can scale to meet the challenges in a de-perimeterised operational environment. The question was answered positive in the same paper, whereas they confirmed that the scale and operational complexity would grow. This was further explained in (Henry S. Teng 2008) as follows:

> *"Control points that were centralised and external to applications and systems will change (end-points have shifted). The shift in control points will create new scenarios of controls that are more application centric and data protection centric. Reliance and assumptions of controls over traditional internal components such as a WAN or LAN, may no longer be relevant or appropriate (audit scope changes). A sampled assessment of decentralised components may not give a clear picture of the overall IT control environment (partners spread spyware, business boundary and IT boundary). The focus and importance of core IT systems may need to change – for example, increased reliance on Data Centre, client and application controls. Additional foundation services (Identity, Audit, Monitoring) may need to be included in the scope of future audits."*
>
> (Henry S. Teng 2008)

If a de-perimeterised company and its auditors would not consider the change of scale and operational complexity, then multiple risks including regulatory compliance risks would raise. This would also affect business Risk Management, expensive costs in light of compliance to the US Sarbanes-Oxley Act.
They also said in that same paper that they believed that the Jericho Forum Commandments would not have any strategic impact on the underlying IT audit control frameworks such as the mature framework as COBIT.
The situation of the perimeterised company is even worse. According to (David Lacey 2006) the organisations would have traditionally underwritten the inconsistent implementation of technology controls in their internal framework by building and maintaining a strong external security perimeter. This will become quite troublesome in

> *"a landscape of increasing threats, vulnerability and regulatory compliance demands, there will be a strong need for evidence that adequate and appropriate governance of Information Security has been implemented and continues to operate across the scope of the organisation and infrastructure involved"*
>
> (David Lacey 2006)

In (Forum 2007d) we found the following changes that need to be considered from a tactical perspective:

- *A sampled assessment of decentralised components may not give a clear picture of the overall IT control environment (partners spread spyware, business boundary and IT boundary).*
- *The focus and importance of core IT systems may need to change – for example, increased reliance on Data Centre, client and application controls.*
- *Additional foundation services (Identity, Audit, Monitoring) may need to be included in the scope of future audits."*

(Forum 2007d)

A little counter research: COBIT and de-perimeterisation

If we look at intermezzo 4 in paragraph 2.4, then we can see that one could have some doubts around the idea of the absence of a strategic impact on COBIT by the concepts of the Jericho Forum. Some of the Jericho Forum Commandments does not seem to be covered at all by the Control Objectives from COBIT. Furthermore, the Jericho Forum Commandments that do seem to have some cover by the Control Objectives are not guaranteed to be covered completely. Some of the control objectives and processes such as DS2 for instance, do not cover the needs for the management of third party services as we will see in section 3.6.6.

Recommendations by the Jericho Forum:

They encourage the IT audit community to establish needed standards, guidelines and solutions that are clearly linked to the management of business risks when embarking on a journey towards business agility in a de-perimeterisation computing environment. (Forum 2007d)

They also identified a set of key challenges and next steps in which they described what kind of challenges an organisation would face when moving towards a de-perimeterised environment. Some of the most important on the field of auditing would be:

- Developing applications that are Internet enabled and take advantage of security controls such as transport layer security, authentication and authorization controls.
- Relying more on end-points in the network to protect themselves using patching, firewalling, anti-virus technologies.
- Revise the following IT audit aspects:
  - *Audit Planning:* the auditor will have to understand the strategy that the organisation is following and where the organisation is along its roadmap. Planning the audit of a de-perimeterised environment is just as important as conducting the audit itself. Because of its decentralised nature, auditors choosing inappropriate systems and controls may miss core foundation systems or waste time with inappropriate systems.
  - *Audit Scope:* the appropriate systems, environments and applications should be covered to meet business audit and audit objectives. Traditional centralised services may not be appropriate, if decentralised controls have been adopted. In addition, the following core foundation capabilities will need to be covered in the scope of an audit in the future: (Forum 2007d)
    - Authentication and authorisation services.
    - Time stamping.
    - Monitoring and auditing.
    - Encryption in transit and storage including data fields.
    - End-point security policy - firewalls, anti-virus, anti-spyware etc.
    - Application security controls such as transaction and workflow related.
    - Security at entry points such as vpn's, remote users, wireless users.
    - Third party communications.
    - Trust relationships with external parties - business partners, suppliers, customers.
    - Data centre controls / SAS 70.
    - Management of outsourced providers.
  - *The audit assumptions:* The audit assumptions need to be revised. We might have thought in the past that we could rely on centralized controls and audit these, yet in the de-perimeterised world, we need to check additional decentralized controls such as those

at end-points as well (on an individual basis). Furthermore, the old assumption of having the internal network secured and out of scope from application audits should be revised to something as: *"The internal network is or could be semi-public or public and as such all applications need to assume that the internal network cannot be fully trusted"*. Another assumption, taking a sample of systems and applications is representative of the IT environment, does no longer hold as well, the new assumption should be *"The scope and scale of audits may need to expand to factoring centralised and decentralised points of control".* These three revisions of audit assumptions are coming from the following three shifts: (Forum 2007d)

- IT controls will have to be moved towards end-points such as Data centres, applications, and clients. (Forum 2007d)
- The organisation's internal network may no longer be truly internal – several business partners, third party suppliers and other users may have access to the network. (Forum 2007d)
- Each system and application will have a combination of centralized and decentralised IT controls. Controls will be built closer to the applications and users themselves. (Forum 2007d)

- *Performing the audit:* When conducting the audit, the auditor will need to identify where controls can be relied upon from a centralised and decentralised perspective. Checklists for effective IT audits are to be developed that will take into account of balancing the business context served by the IT environment and associated IT controls for proper value and assurance. (Forum 2007d)

Furthermore, the following recommendations have been found in (David Lacey 2006):

- **New mixture of best practices:** The current standards and best practices need to be combined to get an effective standard for Regulation, Compliance and Certification. The following elements have been mentioned:
  - *A Code of Practice* and assurance process for Information Security Governance across the scope of the shared organisation/infrastructure. (The ISO 17799 standard and certification process meets this need.)
  - *Approved security implementation standards* for supporting infrastructure - either generic for a platform type (e.g. desktop, server, firewall, etc.) or specific for a particular release. Many such specific standards have been developed by individual organisations or have been published within private security circles (e.g. ISF). Nevertheless, new work is needed to establish a generic set of profiles that can be recognised by all organisations.
  - *Assurance standards* for technology components, critical to the security of the supported information systems. (The Common Criteria meet this need.)
  - *Real-time monitoring processes* that can detect and report potential security vulnerabilities or breaches of security.
  - *A management framework* that brings together all of these components into a guide that explains how and when to apply particular schemes and components.
- **New standards such as the Common Criteria:** Since organisation will be unable to maintain up-to-date operational security standards for every single technology item that may be used to execute its business. Therefore, organisations must move towards utilising standards such as Common Criteria to define generic protection profiles that can be applied across different categories of technology, followed by service providers and relevant external parties, and easily extended to meet emerging innovations and threats. In addition, it will be necessary to provide node/end-point governance around such protection profiles.
  - *Common criteria need customizations*: The Common criteria allow one to get the maximum possible flexibility of specification. It uses building blocks that specify components of security solutions or development/test approaches in a technology-independent way. Some components can be customised to particular requirements. It is also possible to develop new components. There is a common misconception that the Common Criteria are bureaucratic and costly to follow. This is certainly true of some

existing standards and evaluation methods but the Common Criteria also allow low-cost, non-bureaucratic standards to be built if that is desired. The Common criteria can be exploited to identify common component types and then develop a standard security functionality standard for each component. Components would include the following:
  - The access device - the equipment a person uses to access a computer system.
  - The server device - the equipment an automated service executes upon.
  - The authentication service - to authenticate users, organisations, devices or services.
  - The authorisation service - to authorise users, organisations, devices or services.
  - The audit service - to maintain and query a record of events. This consists of at least three sub-services: an agent that records events; a repository that stores events; and a query/analysis service.
- **New governance model:** Revising the different aspects and using the new elements should create a new governance model

(David Lacey 2006)

IT-Audit and the Jericho Forum Commandments:
If we look at the JFCs then we can give the following recommendations around the IT-auditing standards, systems and controls that will be used in a de-perimeterised environment, then we can find the same recommendations as already made in the papers: the controls need to be appropriate (JFC1), the system should be scalable (JFC2), assume the context (be prepared for more complexity) (JFC3) and use an open and inherently secure standard(JFC4).

Concluding:
Looking back in this section, we can summarise it as follows:
- **IT audit is about** the formal verification and validation of the quality and effectiveness of IT controls to support the overall business control objectives. There are different types of auditing that could be applied such as operational audits, compliance audit, administrative audit and the information security audit.
- **Issues around IT audit in a de-perimeterised environment:** The following issues have been found around IT-audit in a de-perimeterised environment:
  - *Current controls in perimeterised environment are inconsistent:* The current controls in a perimeterised network would be inconsistent due to the building and maintenance of a strong external security perimeter.
  - *There is a tactical impact on terms of scale and complexity*: The complexity and scale of the audit will rise, since there are not just only centralized, but also decentralized controls to audit. Furthermore, the current controls over traditional internal components might not be positioned as today. Even more services might need auditing besides the standard services of today.
  - *Strategic issues:* research has shown that the current governance framework COBIT will not support the Jericho Forum Commandments, which means that additional frameworks or components will be necessary.
  - *Impact of issues will be costly and risk-raising:* If these issues remain, then multiple risks will grow and costly problems might appear in terms of non-compliance with SOX.
- Recommendations around IT audit in a de-perimeterised environment:
  - *Develop new applications*: Developing applications that are Internet enabled and take advantage of security controls such as transport layer security, authentication and authorization controls.
  - *Rely more on end-points*: rely more on end-points in the network and protect them properly.
  - *Revise the following IT audit aspects:* The following aspects should be revised: the planning, the scope, the way that the audit is performed and the audit assumptions around the controls, the network and the system sampling.
  - *Create a better mixture of best practices:* The current standards and best practices need to be combined and further developed: ISO 17799, the Common Criteria (which will need

Counter research holds even voor v2.0 release

The new revision of the COA framework still holds to the old assumptions on the field of stratetical impact and the use of COBIT. However there are no new valid arguments found that our counter research would not hold anymore, there just seems to be a difference of the interpretation of the strategical approach.

further customization, standards around real time monitoring, approved security implementation standards for supporting infrastructures (which still have to be partially developed) . All of this should be combined to build a management framework that brings all of these components together.
    - *Create a new governance model*: all of this should lead to a new IT Governance model, which will help to perform IT-auditing in a de-perimeterised environment.
- The usage of COBIT:
    - *Conflict between Jericho Position Papers and our own findings:* Even though the Jericho Forum stated in their Position Papers that COBIT could be used, since there is no strategical impact on COBIT by the concepts of the Jericho Forum, we have found otherwise: COBIT cannot sustain with the JFCs, so additional work will have to be done on this field.

The new IT-auditing framework should definitely be part of the COA framework, as we will see in chapter 3.

COA V2.0 UPDATES:

The following relevant additions have been made in the COA 2.0 revision:

- **Additional findings on the field of IT-audit**, in: "IT Audit and Compliance", which can be summarized as follows:
    - *The impact of de-perimeterisation on TOD(Test of Design) and TOE(Test of Effectiveness):* one will need the ability to demonstrate the same risk-based control quality in a de-perimeterised environment as in a bounded one.
    - *COBIT would still hold from a strategic impact viewpoint:* The process sets and the principles or qualities of the control objectives are still usable as there would be no strategical impact of the Jericho Forum Commandments on them.
    - *A set of challenges for de-perimeterising companies*: a set of challenges have been defined which mostly consist of the usual challenges for de-perimeterisation itself, such as thinking of the internal network as a semi-public one, etc..
    - *Elements that should be in scope of the audit*: the following elements have been named: authentication & authorization services, time stamping, monitoring and auditing, encryption, end point security policies, application security controls (i.e. workflow related), security at entry points (i.e. VPN's, remote users, etc.), third party communications, trust relationships, data centre controls/SAS 70, management of outsourced providers.

The publications can be found at: https://www.opengroup.org/jericho/publications.htm

2.6.11. Summarising: the Jericho concept

Looking back on this paragraph, we can see that we have discussed a broad set of topics. All of them are important to be capable of de-perimeterisation and collaborating in a more dynamic and secure way. If one wants to be capable of using the COA framework to its maximum value, then he should understand the concepts that have been discussed here. That is why we will get back to this paragraph in chapter 3.

However, it is important to realise that one can get lost easily due to the broad range of Jericho related topics and their respective details. That is why the COA framework will be extremely important. It will allow one to use a complete and systematic approach for realising the Jericho concepts in the information architectures of today.

We will try to summarise the concepts we have discussed as follows:

- **The Jericho concept or "de-perimeterisation"** is based on the story of the fall of Jericho. Its main focus is de-perimeterisation that is about a change of focus from large perimeter-based defence to focussed information security. The basics "how-to's" of the concept are

summarised in the 11 Jericho Forum Commandments which can be found in appendices A1 and A2. The most important themes are the need for trust, data protection (conform the asset at risk), survival in the hostile world of the internet and the need for identity, management and federation.[83]

- **The Jericho Forum** is a group of CISOs that has a vision focussed on enabling business confidence for collaboration and commerce beyond the constraint of the corporate, government, academic and home office perimeter. Its mission is to act as a catalyst to accelerate the ideas inside the vision. So basically, to be an advocate for de-perimeterisation.
- **Reasons for de-perimeterisation**: there are multiple reasons such as cost reduction, increasing flexibility, the insecure boundaries, the need for securing data itself and better business connectivity.
- **Protocols and standards should be open and inherently secure:** the protocols and standards that should be used for the technologies and solutions inside a de-perimeterised enterprise should be open and inherently secure. However, the chosen amount of security measures and the chosen protocols should be appropriate to the asset at risk. All of this should increase the trust in entities such as devices.
- **Security mechanisms, services, protocols and other measures:** many security services (entity authentication, data origin authentication, access control, confidentiality et cetera), mechanisms (encipherment, digital signatures, access control et cetera) and inherently secure protocols (SSL, IPSec et cetera) have been discussed. Application level devices such as firewalls, proxies and other concepts such as message protection, have been discussed as well. All of them are publicly available, yet some issues will arise in the use of them.
- **Policies in the Jericho Concept** are mostly focused on information access and information security. Current standards such as the ISO27000 need additions in order to fully cover the concepts of the Jericho Forum Commandments. Additionally, there is a need for a more powerful, scalable, auditable and flexible policy management system, which should use a XML based language. The COA framework should be or hold the answer to the recommendations and issues on this matter.
- **Data classification I**n order to apply JFC1, data classification is a necessity. There is a need for temporal classification, a fine-grained information security infrastructure and a language such as XACML. The traffic light protocol and the currently under discussion automated security classification can both be a great help.
- **Data protection** should be based on encryption and Digital Rights Management, Furthermore, measures such as metadata in the body with the capability of blanking out certain parts are recommended as well. There should also be an open standard for Enterprise Information Protection, to be capable of exchanging data between current Enterprise Information Systems.
- **Data privacy** will need segregation of duties and a good approach that allows the subject to be in control of the data about him.
- **Identity Management** consists of technologies and business process, which are aimed at managing the identities inside and around the company. An identity is a fundamental concept of uniquely identifying an object within a context. There are various identity approaches and the identity in favour of many companies is the federated identity, while the user centric identity is the best solution from a Jericho perspective. Other important concepts are those of authentication, verification, revocation, authorisation and accounting. It is important to realise that the objects that are managed could be devices, organisations and/or human beings.
- **Trust** is a very important subject in the concepts of Jericho. It is tied to many other concepts such as collaboration, Identity Management et cetera. It is defined as “a particular level of the subjective probability with which one assess that another entity or a group of entities will

[83] The commandments are very important for all of the following concepts, see their respective sections for more details.

perform a particular action before one can monitor the action. This action will be beneficial or at least not detrimental to the one that trusts the entity or entities". It has multiple sources, which need to be re-evaluated continuously. This also counts for digital trust.

- **Trust management:** in order to manage trust, multiple models have been created, we have seen some promising ones such as ReGreT and many others. It is clear that there is a demand for an architecture in which a trust broker can be used to manage and establish trust between parties. There are various types of trust brokers discussed and a set of recommendations and requirements for a trust broker framework have been made.
- **End-point security** is about raising the level of inherent trust in computing devices to a point where all the devices involved in a transaction meet the criteria of trust for that transaction. That level of trust is based on the status of the firewall, encryption standards and other aspects of the system hardening. All end-points will have to follow certain security postures in their zone. A set of requirements and recommendations have been made on the field of end-point security such as requirements for interoperability, open standards, security requirements and recommendations about the usage of multiple end-points in multiple zones, two way trust, the use of the trust broker architecture for end-point security, the use of automation and security agents et cetera.
- **IT audit** is about the formal verification and validation of the quality and effectiveness of IT controls to support the overall business control objectives. The Jericho concepts will have a major impact in terms of scale and complexity. We have discussed multiple recommendations such as the creation of a new governance model, the developments of new applications et cetera. However, there is still a point of discussion left: as the Jericho Forum believes that COBIT will suffice as a framework, we believe otherwise.

2.7.Summary and overview

All of the backgrounds have been researched and discussed for as far as possible with the available resources.
Most of the basic backgrounds will have a lot of influence on the COA framework: many issues and questions that rise on the field of SOA collaboration for instance, will have to be resolved by the use of the COA framework.
Other principles such as the theories around Collaboration will have a major impact on how the COA framework can be applied under certain circumstances.
Even more important are the concepts of the Jericho Forum, which we will see again in the COA framework, yet more integrated than in the current situation.

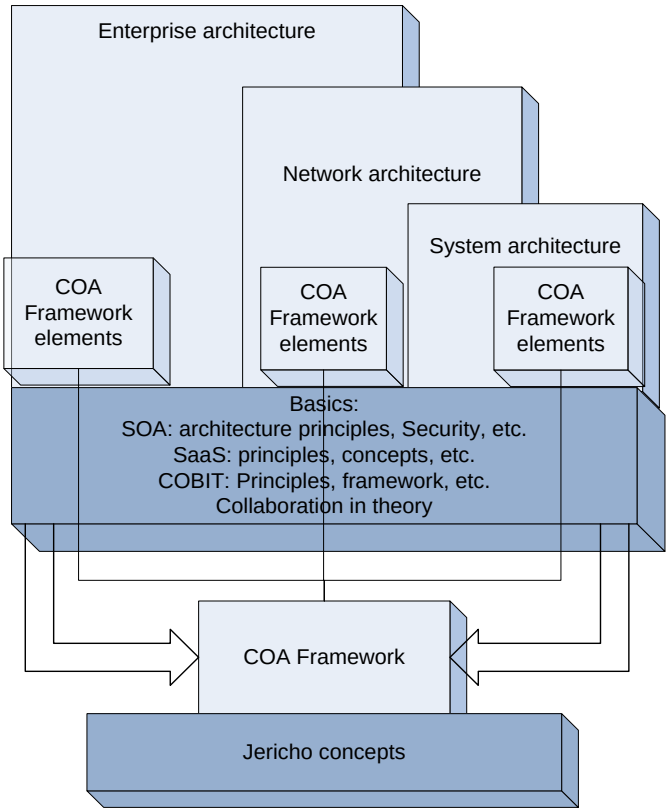


Figure 32: Relationship between the different elements, marked areas have been covered in this section.

Even though these basics and Jericho concepts are intriguing by themselves, this chapters only goal and purpose is to investigate the basics and the Jericho concepts in order to be capable of creating the COA framework, which we will see in chapter 3. In this chapter, we have discussed the following topics:

- **Service Oriented Architecture**: as the COA framework will be build upon any existing SOA (an architecture based on services), it becomes important to understand what it is and what concepts it holds. By understand what SOA is and what it brings, we can see the value of the COA framework and understand what we could expect in a SOA when we try to apply the COA framework.
- **Software as a service**: We have discussed the concept of SaaS and its current issues on the field of security and trust. SaaS is one of the concepts that could gain many benefits from the COA framework and is therefore very interesting.
- **Control objectives for Information and related Technology:** we have looked at the COBIT Framework and IT Governance with its related frameworks and measures. We have seen that much work needs to be done on the field of (IT) Governance within the COA framework and we have selected the DS2 process to experiment with as a start. So more research should be done on this field in relation to the COA framework in further works.
- **Collaboration**: we have looked at collaboration and some of the various theories around it. We foresee many issues that will rise while implementing the COA framework based on the collaboration theories. However, we also acknowledge that there is still much to research be done in further works on the field of collaboration and the COA framework.
- **The Jericho Forum and their concepts:** we have studied most of the Jericho concepts that will be necessary for the COA framework. Many of those concepts still contain loads of issues and recommendations that still have to be taken into account when we will detail the COA framework.

3. The COA framework

3.1.Introduction

Now that the basics have been covered in chapter 2, we should have all the necessary knowledge for detailing the COA framework. The framework is based on the Jericho concepts and is highly influenced by the basics, as one can see on Figure 33.

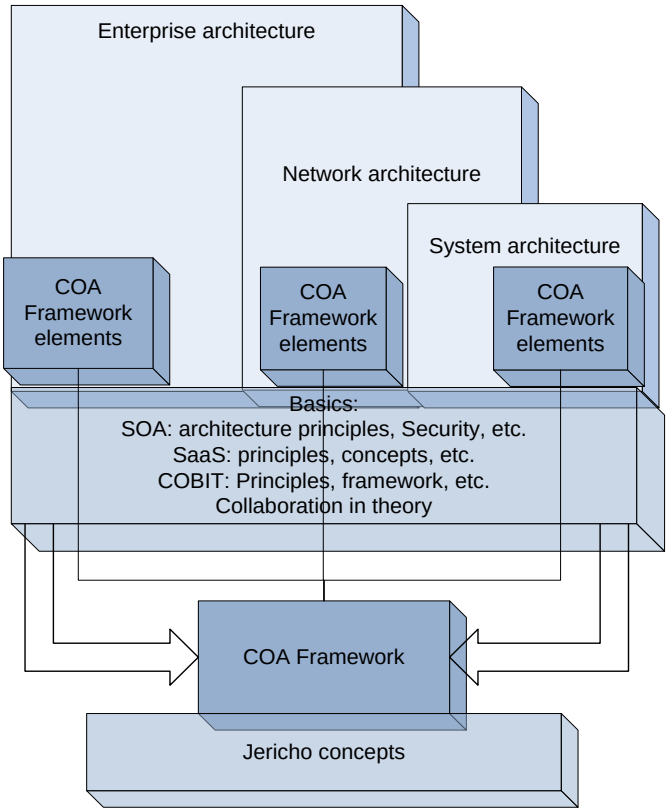


Figure 33: Thesis structure, marked area will be covered in this section. Marked transparent areas have been covered.

The COA framework should never be seen out of the context of the covered basics and the Jericho concepts. It is assumed that the reader is familiar with the context elements that underpin the COA framework.[84] In this chapter, we will discuss the following:

- In paragraph 3.2 we look at the current situation in which the COA framework is a necessity. We observe multiple concepts such as the current developments inside the Jericho Forum and the need for the COA framework as has been expressed in the media. This sets the stage for paragraph 3.4 and the answer to research question number two.
- A general introduction to the COA framework is given in paragraph 3.3. The basic overview of the framework provides insights in the relationship between its components and the purpose of the components themselves. This partly addresses research question one.
- The value of the COA framework is discussed in paragraph 3.4, which will be expressed in terms of SLATES and the value that is added in terms of security and collaborative possibilities. Research question two will be addressed in this paragraph.
- Paragraph 3.5 further elaborates on the COA principles. We will detail the principles and discuss what the principles mean, knowing what we know from chapter 2. This partly addresses research question one as well.
- The processes of a COA are further detailed in paragraph 3.6. The papers that are currently under development and the results of the background research will be used in order to give a short description and put out a set of requirements and recommendations for the COA processes. This will also partly address research question one.
- The services of a COA are further detailed in paragraph 3.7. The papers that are currently under development and the results of the background research will be used in order give a short description and to state a set of requirements and recommendations for the COA services. This partly addresses research question one.

[84] If you are not, read chapter 2.

- After implementing the COA framework, one can use a set of quality attributes to test whether an implementation has been successful. These factors will be discussed in paragraph 3.8. This partly addresses research question one.
- The technologies to be found in a COA are defined in paragraph 3.9. This partly addresses research question one.
- Paragraph 3.10 maps the relations between the Jericho Forum concepts and the COA framework.
- Which components of the framework can be used in certain cases and architectures will be discussed in paragraph 0. SaaS, the enterprise architectures, the network architectures and system architectures will be taken into account for using the COA framework. This addresses research question two.

This chapter will be summarised in paragraph 0.
The route from paragraph 3.3 to 3.12, excluding 3.10, is visualised in Figure 34. First the framework and then (1) its use will be discussed in general (paragraph 3.3), from there we detail the framework (2) (paragraph 3.4 till 3.8) and then try to detail the application and/or adoption of the framework (3) (paragraph 3.11).

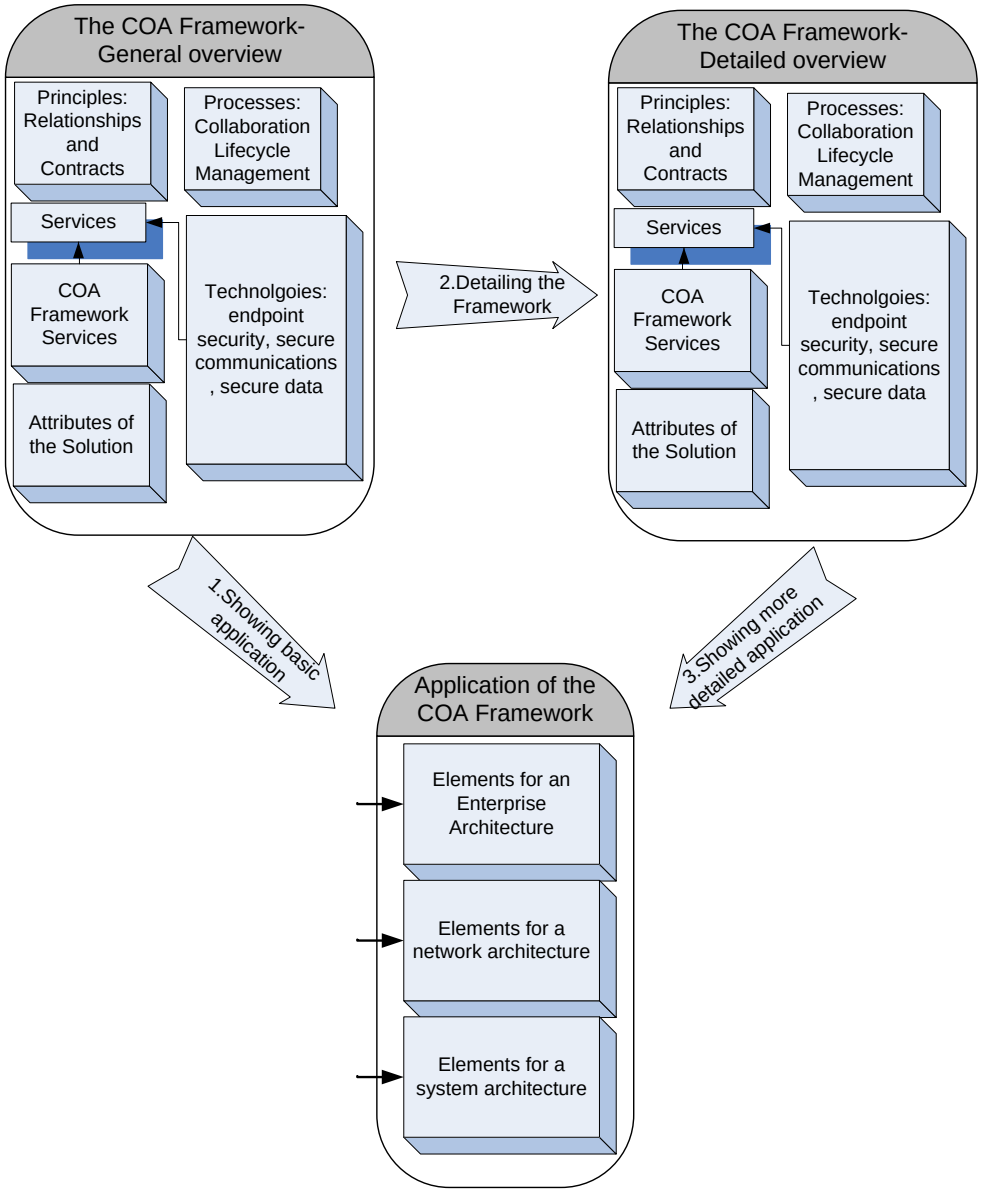


Figure 34: Overview of framework detailing process.

However, one should realise that most of the promised position papers surrounding the COA framework are still missing. That is why most of the work here is based on e.g. inferred from the basics and the Jericho concepts.

A final remark should be made about the work that will be presented in this chapter. There is a set of sources, residing inside the Jericho Forum that has been called the Challenge Papers. These have emerged too late (even though they have been written long ago) to be used in this thesis. If one would like to have a full picture, these papers ought to be looked at.[85]

3.2. Looking back: the current situation with its complications

3.2.1. Introduction

The COA framework itself is a framework that can be used as an addition to current information architectures (Forum 2008e). Therefore, before one can understand the value, the purpose and the contents of the framework, one will first have to understand the current situation in the SOA-driven markets of today. In other words, this paragraph allows us to gain more knowledge of the current situation in order to be capable of answering research question number two.
As the COA framework is a solution based on the Jericho concepts, one will have to understand why the Jericho concepts alone are currently not sufficient.
Both these topics, the current situation in the markets of today that creates the necessity for the framework (section 3.2.2) and the current status of the Jericho concepts (section 3.2.3) will be discussed shortly in this paragraph. It will set the stage for the introduction into the COA framework itself. The paragraph will end with a short summary of what has been discussed (section 3.2.4).

3.2.2. The current situation in the markets of today

In order to analyse the current situation, both data from the sources in chapter 2 and new data from Gartner will be used. However, we do not pretend to give a thorough and complete picture. It will be only a sketch of what we have seen during this research project. There are several issues at hand:

First, we can see that SOA is having a major influence on today's business. Many companies are implementing it thanks to the promised reuse, flexibility and lower costs. Research bureau Gartner has predicted the number of companies doing so will grow to 80% of all companies in 2010 (Natis 2006). Since there are no strict rules of how one should see SOA or how it should be implemented, every implementation is considered legitimate. This makes SOA a tempting architectural approach for many companies: each can perform its own implementation allowing the creation of unique architectures. The differences between the architectures making them unique will also make it harder to let them collaborate securely. Even though SOA allows more dynamic collaborations, there are still many crucial security issues left. Many have tried to create collaboration frameworks for SOA, yet none of them have become successful. The security issues at hand, as seen in section 2.2.3, remain, and so does the need for a framework to cover many issues in terms of collaboration and information security (see paragraph 2.2 for more details).

[85] The papers can be found at http://www.opengroup.org/jericho/protected/documents.tpl?CALLER=doc.tpl&dcat=19 , visited at 13-11-08.

(Jothy Rosenberg 2004; Ivar Jørstad 2005; Bingnan Xiao 2006b; W. T. Tsai 2006; X. Zhou 2006; Hutinski 2007a; Ismail Khriss 2007; Ohrstrom 2007; Sanjeev Kumar 2007; W. T. Tsai 2007; W.T.T sai 2007; Herwig 2008; Liam O'Brien 2008; Stan 2008b; Surya Nepal 2008)

Second, the need for collaboration has almost exponentially expanded. As we have seen throughout the research in chapter 2, many pieces of literature show that collaboration has become an absolute necessity (see paragraph 1.1, 2.5 and 2.6) in order to survive and thrive in today's markets. Creating the appropriate measures in terms of collaboration and security has thus become elemental (see paragraph 2.3, 2.5 and 2.6 for more details).
(Boonstra 2002; Philip Kotler 2002; Forum 2005; Forum 2007f; Forum 2008e; Joziasse 2008; Ralph Welborn 2008)

Third, that need for collaboration has been translated to many different collaboration models (see paragraph 2.5) and relationships which need security measures or some form of reassurance, to gain confidence and trust in them (see SaaS for instance: paragraph 2.3 and the need for trust in section 2.5.5 and 2.6.8). However, many of the current most effective measures are still time-consuming.
(Nicolas Gold 2004; Markku Sääksjärvi 2005; Forum 2006f; Marle 2007; Bruning 2008a; Bruning 2008b; Demarteau 2008; Dirk Hanenberg 2008; Leijden 2008; Metsaars 2008b)

Fourth, the current collaborative relationships require interconnectivity, intercompany IT-alignment and governance. This interconnectivity has to be obtained through current perimeter defence systems, which create new holes and vulnerabilities. We have also seen that the current IT-alignment and governance methodology are flawed, or at least quite cumbersome. (see paragraph 2.4 and 2.6).
(Forum 2005; Forum 2008e)

Fifth, the information itself needs better protection. As information containers (such as documents) are compromised, they easily spread out "in the wild" e.g. the internet, where there is often no protection at all. The same goes for personal identifiable information. This makes it even harder to manage one's own privacy (see paragraph 2.6 for more details).
(Forum 2005; Forum 2006g; Forum 2006e; Forum 2007e; Forum 2007c)

Finally, most of the interconnection related observations that have been mentioned here are part of the de-perimeterisation as described in section 2.6.3: The current perimeter based approach is flawed because there is no perfectly secure boundary, we need to secure the data itself. That is because of a number of reasons such as the online threats are becoming ever more sophisticated and the business which is requiring more connectivity through that perimeter, a cost reduction of the current security endeavours and more flexibility in terms of IT and resources.

This sketch shows some of the problems that the Jericho Forum has seen and has described by their own means in different position papers and other media (see paragraph 2.6 and the next section for details).
However, the problems sketched and noticed by the Jericho Forum have also been noticed by Capgemini and multiple other organisations. Only a real solution would be the next logical step.

3.2.3. The current status of the Jericho concepts

Having shortly outlined the current situation, a description of the actions by the Jericho Forum to resolve these matters must be given, which allows the understanding of the COA framework as an absolute necessity.

The Jericho Forum had its first gathering in 2003[86] and ever since, it has drawn a lot of media attention. Multiple meetings[87] to which an ever-growing number of members attended, resulted in the release of many articles among internet sites[88], as well as the release of a set of publications consisting of position- and white papers[89]. All of the released material focussed on different aspects of de-perimeterisation and the issues shortly addressed in the section above.

Not only the Jericho Forum drew this attention; other companies, often members of the forum such as Capgemini, have released multiple articles and publications around the Jericho concepts and related issues[90].

The Forum and its members tried to address all of the abovementioned issues. They have shown the Forum understands the sketched issues and that it wants to give an answer to those issues. Most of the issues could be partially resolved by de-perimeterising.
However, none of the publications, meetings, and writings have currently provided a true understanding of how one should de-perimeterise. Most papers gave multiple reasons for de-perimeterisation and showed what is wrong with the technology, architectures and business processes of today. Furthermore, recommendations for a solution have been made, but they often went as far as setting up a combination of requirements for only part of the complete solution without going into details, since they are generally related to a context or architecture.
The concepts of the Jericho Forum have been addressed and found their way to the masses. However, those masses still lack a solution and a roadmap to that solution that covers these concepts.

This problematic state went on until March 2008 when the COA framework was introduced as the solution for de-perimeterisation. While writing this thesis, the COA framework itself is still far from finished. The first position paper promises enormous value (see also paragraph 3.4) but the other promised papers are still in, or awaiting, production.
Even though the framework is not finished, many have described the necessity of it in terms of value around SLATES and security (See also paragraph 3.4)[91]. Organisations such as Capgemini, Domus Technica, Shell, Boeing and Ely Lilly would like to see the framework finished to a useable and implementable state.

---

[86] Source: http://www.opengroup.org/jericho/about.htm, visited at 15-10-2008.

[87] Source: http://www.opengroup.org/jericho and various sub-sites, visited at 15-10-2008.

[88] Source: http://www.opengroup.org/jericho/newses.tpl?CALLER=index.tpl, visitied at 15-10-2008.

[89] Source: http://www.opengroup.org/jericho/publications.htm , visited at 15-10-2008.

[90] Sources:
http://www.nl.Capgemini.com/resources/news/Capgemini_brengt_nieuwe_manier_van_informatiebeveiliging_in_de_praktijk/,
http://sites.vnuexhibitions.com/sites/bezoekers_infosecurity_nl/nl/page.asp?module=pages&type=item&id=16515,
http://www.jobbingmall.nl/INTL/JobSeeker/Jobs/JobDetails.aspx?job_did=J8D3B07248F45V7JSPD&cbRecursionCnt=1&cbsid=188ff4ebd07f4506bbc2d32ab29a4508-277361843-R7-4&ns_siteid=ns_nl_g_jericho_Capgemini,
http://www.Capgemini.com/ctoblog/2006/07/the_walls_of_jericho_1.php, see also
http://www.google.nl/search?hl=nl&q=jericho+Capgemini&meta= , visited at 15-10-2008

[91] Some of the sources: http://www.computerweekly.com/Home/tags/collaboration-oriented.htm,
http://www.computerweekly.com/Articles/2008/04/08/230188/rsa-2008-collaboration-oriented-architecture-key-to-web-2.0.htm, http://srmsblog.burtongroup.com/2008/05/jericho-forum-a.html,
http://www.networkworld.com/columnists/2008/071008-jericho-collaboration.html , al visited at 16-10-2008.

3.2.4. Concluding: the necessity of the framework

We see the framework is a necessity in order to resolve the issues that have been sketched in section 3.2.2 and throughout chapter 2, with de-perimeterisation as one of the most challenging problems to resolve. As the current answer from the Jericho Forum in terms of its position papers and publications insufficiently addresses the issues that have been sketched, many hope the COA framework itself soon becomes the answer. Companies such as Capgemini, Domus Technica, Shell, Boeing and Ely Lilly would like to possess a ready-to-use version of the COA framework, to address their security and collaborative issues.

3.3. An introduction to the COA framework: general overview

3.3.1. Introduction

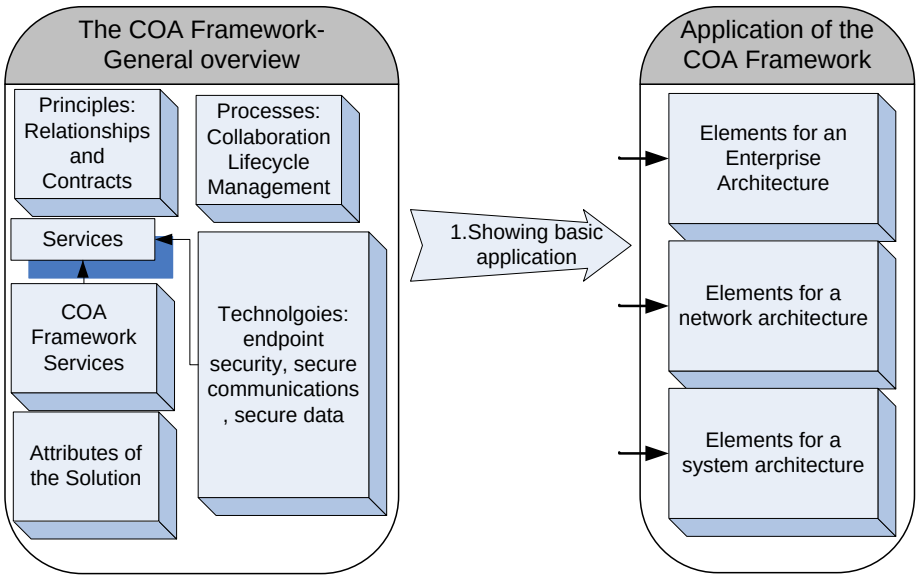


Figure 35: Overview of paragraph 3.3.

As the necessity of the framework is clarified previously, the COA framework must be properly introduced.
We will answer the research questions based on available knowledge.
After this we further detail the COA framework and its value based on research from chapter two in paragraph 3.4 to 3.8, and its application in paragraph 0.
The process of the current paragraph is visualised in Figure 35. This process allows a partial answer to research question one (What is the Collaboration Oriented Architecture Framework?), two (Why is it important?) and three (How can it be adopted?). Most of the work here is based on (Forum 2008e) and the position papers that are still in concept within the Jericho Forum.
The following subjects will be addressed in this paragraph:

- The COA framework itself will be observed in section 3.3.2, where most of the information is based on the released position papers.
- The purpose of the framework in general will be discussed in section 0.
- The application and/or adoption of the framework will be discussed in section 3.3.4.

We conclude this paragraph in section 3.3.5 summarising the findings and partially answering the three research questions.

3.3.2. The Architects' View and its components

The COA framework will be discussed in this section, based on (Forum 2008e)[92]. This provides a partially answer to research question one ("What is the Collaboration Oriented Architecture Framework?").
The COA framework is visualised in Figure 36 and consists of four groups of components: Principles, Processes, Services and the attributes of the Solution (Forum 2008e). As one can see at the architect's view, a fifth group can be added to them: Technologies.
Observing the groups separately:

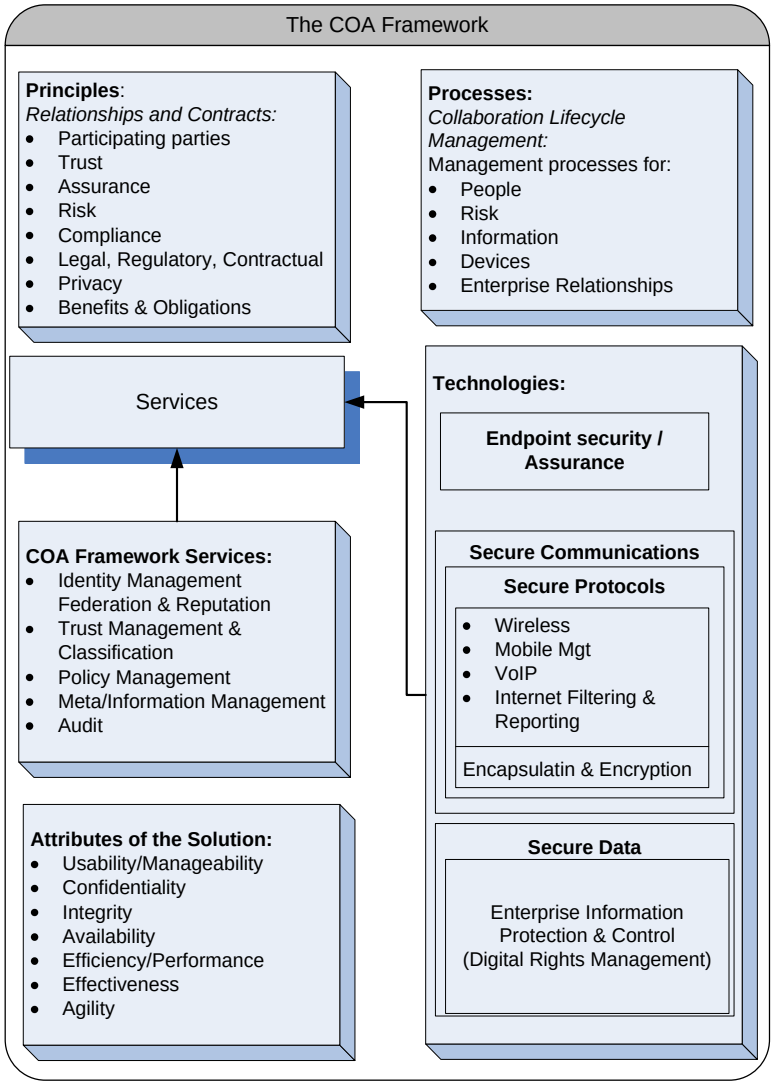


Figure 36: The Collaboration Oriented Architecture Framework from an architects' view (based on (Forum 2008e)).

[92] The other papers are still in concept and will be used in the more detailed descriptions. Later on in this chapter

The COA framework:
The COA framework is created in order to change the current information architectures into a Collaboration Oriented Architecture (COA), which differs from traditional architectures in terms of it not aiming at securing organisational borders, and then the network, reinforcing a 'perimeterised' perspective. The COA framework instead defines the key components within which interoperable, secure solutions can be provided to meet the needs of a business in terms of interconnectivity. Implementing the COA framework will change the focus of the information architecture and create a COA of it.
A COA enables provision of IT systems that are secure in a global networked world, able to keep pace with the growing threats and the business need for faster and more flexible collaborative business arrangements. These range from outsourcing to joint ventures, from merger today to divestment tomorrow, all within a global working, global manufacturing and global procurement environment. (Forum 2008e)
The adoption of the COA framework is based on JFC4-JFC8 and will enhance the information architectures, by increasing the emphasis on the COA framework Principles, by providing a set of services, processes and technologies which allows one to create enhanced accounting logs, gain transparency on the identities, reputations and impact of each individual and device in the collaborative environment et cetera. It will also provide a set of quality attributes as a checklist to see whether one has achieved his goal by implementing the elements of the Framework.
Finally, the COA framework is the architectural solution as a response to de-perimeterisation. It allows one to re-perimeterise or de-perimeterise safely and securely.
(Forum 2008e)

Principles- Requirements (must haves) and Constraints (shall not's):
The Principles of the COA framework consist of requirements and constraints. A requirement can be seen in this context as an obligatory element. These are necessary to ensure a safe (set of) collaborative relationship(s) between (multiple) companies that are trustworthy. They also help keeping the relationship safe and trustworthy over a longer period of time. If one does not follow these principles, one becomes less trustworthy and less safe to collaborate with.
Normally, these principles only account for events happening outside the enterprise domain. As the boundary between the outside and the inside fades away because of de-perimeterisation, they will however take care of the inside domain as well.

Figure 37: Principles of the COA framework.

The following principles are part of the COA framework:

- **Participating parties**: The first principle is about knowing with whom and what you are transacting: All components of a transaction chain must be known to the contracting parties at all of its end-points. These components are selected by collaborating parties, during contract negotiations. Collaborating parties are responsible corporate or individual entities whose identities are well- defined and whose activities are controlled by legal, economic, ethical, and technical means. A collaborating party may be a consortium, in which case the consortium must indemnify its members (and provide other economic, ethical, and technical controls) so that other collaborating parties may safely collaborate with the consortium members. In case of individuals, they will initiate interaction through an accredited Identity Service Provider.
- **Trust:** Second, one will have to understand the level of trust/confidence one will be transacting at. This means the collaborating parties have the ability to agree/define appropriate (known) degrees of confidence in the components in a transaction chain, including the environment in which these components are operating.

- **Assurance:** Third, one will have to agree to the level of trust/confidence he will be transacting at before the transaction itself starts. Prior (contractual) agreements between collaborating parties define their obligations to respect each other's intellectual property and to provide adequate technical security during a collaborative transaction.
- **Risk:** Fourth, one should understand the risk within and surrounding the transaction. The collaborating parties may assess any proposed transaction based on the communicated levels of trust with factors germane to the transaction: identity, confidentiality, integrity, availability, location, environment (space it is being used in), data-sensitivity, transaction value, time et cetera.
- **Compliance:** Fifth, one should comply to the rules and regulations of the security inside the collaborative group. Collaborating parties agree to periodic inspections and security audits. The results of these inspections and audits are published within the collaborative group. Non-compliant parties may be sanctioned or expelled.
- **Legal/regulatory/Contractual:** Sixth, the collaborating parties must comply to applicable legal, regulatory and contractual requirements. They must be able to resolve conflicts that may arise, through effective verification and enforcement mechanisms. Additionally, compliance to local, legal and regulatory requirements alone is unlikely to be sufficient to meet all business requirements.
- **Privacy:** Seventh, privacy is a particularly important requirement the collaborating parties must meet. Increasingly, privacy is being defined in legislative safeguards; the consequence of widespread belief in privacy as a fundamental human right. At its root is are customers, suppliers, and employees expecting organisations to use information about an individual ethically so that it is not divulged if it is reasonably considered "private".
- **Benefits and Obligations:** Last is a set of obligations and requirements: Contractual obligations, service level agreements, customer expectations, corporate policy and norms of good corporate citizenship are requirements that need to be aligned and implemented.

(Forum 2008e)

Processes:

The processes inside the COA framework have been designed to provide the collaborating parties with the capability to apply the concepts of Enterprise 2.0[93] safely in their collaboration.

The enterprise 2.0 concept that is used in (Forum 2008e) is based on SLATES:

- **Search**: Discoverability of information drives, reuse, leverage, and ROI.
- **Links**: Using URIs to forge thousands of deep interconnections between enterprise content 24/7.
- **Authorship**: ensuring every worker has easy access to Enterprise 2.0 platforms.
- **Tags**: Allowing natural, organic, on-the-fly organisation of data from every point of view.
- **Extensions**: Extend knowledge by mining patterns and user activity.
- **Signals**: Make information consumption efficient by pushing out changes.

(Hinchcliffe 2007)

Processes:
Collaboration Lifecycle Management:
Management processes for:
- People
- Risk
- Information
- Devices
- Enterprise Relationships

Figure 38: Processes of the COA framework.

This is an easy checklist to see if the tools one is considering have the right essential ingredients for enterprise 2.0 platforms, which are basically web 2.0 applications for the enterprise domain. They can be seen as key transformational elements changing the way organisations do business. A well-implemented COA will maximise the value of collaborations, using various SLATES elements, while managing information risks to an acceptable level. However,

[93] See also: http://blogs.zdnet.com/Hinchcliffe/?p=143 , visited at 18-10-2008

(Hinchcliffe 2007) has argued that a more refined conception of Enterprise 2.0 would be necessary: Freeform, Links, Authorship, Tagging, Network-oriented, Extensions, Search, Social, Emergence, Signals (FLATNESSES). Because (Forum 2008e) has focused on SLATES, we do the same. It could be a good subject to further research: enabling other Web 2.0 content in Enterprise 2.0 safely and secure by the use of the COA framework.

Getting back to the processes, the following have been defined in (Forum 2008e)[94]:

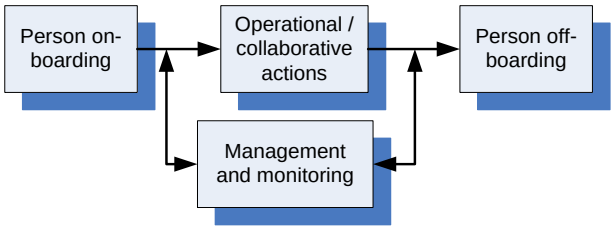


Figure 39: People Lifecycle Management (simplified).

- **People Lifecycle Management:** The People Lifecycle consists of several important events. It starts with a person on-boarding the organisation. Following, his identity, capabilities, capacities and reputation will be managed and monitored. Meanwhile, the person who on-boarded the organisation will execute actions and show a certain behaviour, which will be processed and managed again in terms of reputation, capabilities, capacities and his identity. This will continue to happen until the person is being off-boarded. The cycle then ends. The processes that manage these cycles should also include the management of individuals that are not employees or, more generally, members of the managing entity. Summarising the lifecycle management processes: they take into account the identity[95], personas, capabilities, reputation, and potential impact of each of the individuals.
- **Risk Management:** Risk Management is a cycle as well. One will first have to assess the risk of certain contexts and then take certain reductive actions if necessary. In the meantime, one will have to monitor the risks to see if the situation escalates and additional actions are necessary. After the risk-reducing actions, one will have to assess and monitor the risks again.[96] The Risk Management process consists of processes that identify, classify and manage the information risks involved in collaborations.

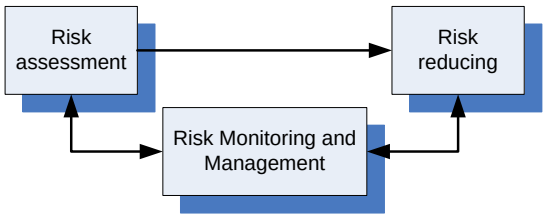


Figure 40: Risk Management (simplified).

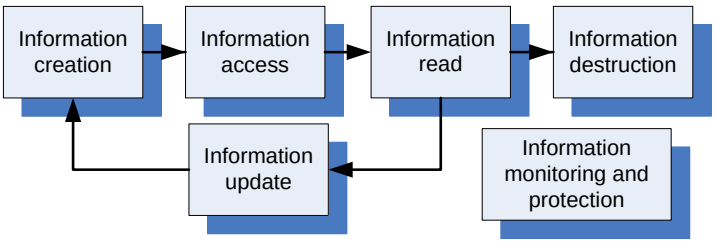


Figure 41: Information Lifecycle Management (simplified).

- **Information Lifecycle Management:** The lifecycle of information consists of the creation, using, updating and the destruction of the asset.[97] In the meantime, it will have to be

[94] The Processes we found here are also referred to as PRIDE (the first letter of each of the processes combined)

[95] The identity is not just the identity, it will also comprise all the attributes of the identity.

[96] One could argue that Risk Management will follow the PDCA cycle as one can see in intermezzo 2. Yet we have chosen to take a simplified approach for now.

[97] One could argue that there will be more steps in a process such as the spreading of the information.

protected against unauthorised actions and monitored at all times. Processes that effectively and efficiently manage the creation, reading, update and deletion of information assets in a collaboration, would include audit, monitoring and information protection activities.

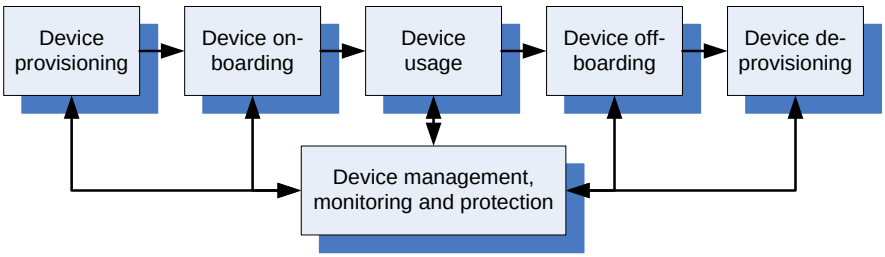


Figure 42: Device Lifecycle Management (simplified).

- **Device Lifecycle Management:** the device lifecycle looks a lot like the people lifecycle. Devices will get provisioned, on-boarded, monitored, managed and eventually off-boarded and de-provisioned. The lifecycle involves processes for introducing devices, identifying and maintaining device trust levels and removing devices involved in collaborations. Removal of devices involves eradication of all information assets from that device. Furthermore, there is the software lifecycle which is embedded in the device lifecycle according to (Forum 2008e).
- **Enterprise Relationship Management:** Processes ensuring that collaborations are managed according to the state of the relationships involved and the value and/or risks they introduce. Initiating, operating, and closing down collaborations emanating from an enterprise would include a means of mapping the critical relationships between all the collaborating parties. Such processes would also have the ability to identify collaborating parties that are endangering the enterprise and rapidly close down offending relationships. The processes would have the ability of identifying the most valuable relationships in order to ensure their appropriate development and protection. Such processes remain valuable during, mergers, acquisitions or divestitures.

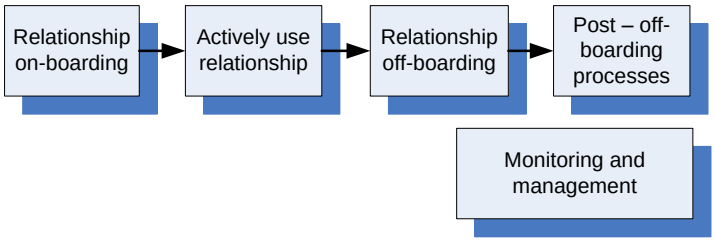


Figure 43: Enterprise Relationship Management (simplified).

(Forum 2008e)

Services:

In (Forum 2008e), a set of services has been specified that should be provided by the collaborating parties or a third party. The one used will have significant ramifications on how the services are provided. The services are focussing on several security issues that have been described in chapter 2. (see paragraph 3.7 for more details)

COA Framework Services:
- Identity Management Federation & Reputation
- Trust Management & Classification
- Policy Management
- Meta/Information Management
- Audit

Figure 44: Services of the COA framework.

The following have been specified:

- **Identity Management, Federation and Reputation**: The credentials of principals (organisations, individuals, systems, devices) and associated attributes required for identification, authentication and authorization decisions, are expressed in a standardised form. These credentials can be validated and accepted by the systems of any member of the collaboration or service providers. Also, the reputation of an internal member will be

managed. The reputation of an external member will be covered with the trust management service.

- **Policy Management**: The collaborating parties, and service providers, have the ability to either jointly or separately evaluate, manage, and implement the policies and rules for authorizing and de-authorizing principals and collaborating parties.
- **Trust management and Information Classification**: The sensitivity of Information Assets is defined with causes of the information risk (i.e. Confidentiality, Integrity, Availability and Authenticity) against a commonly agreed classification model, aligned with risk-based assessment of business impact of an incident or threat. There are identity, legality and temporal components of information classification, all of which are context-sensitive. Trust management itself must be done in cooperation with the trust management framework and especially with the trust broker, which is an essential part of that framework (see section 2.6.8 for more details).
- **Information Asset Management (also called Meta/Information Management)**: Collaboratively shared data is appropriately secured in storage, transit and use, based on the agreed risk and performance requirements for the information contained in the data, as a result of the Classification. Principals accessing the data are identified, authenticated, and authorized. These requirements must be maintained through the complete document lifecycle, from creation to destruction, by appropriate record-management.
- **Audit**: Transfers, storage, and retrievals of collaboratively shared data, and associated business controls, are auditable events. There is a common notion of 'event' across all collaborating parties and systems. Collaborating parties may require each other to conduct spot-audits on individual data objects and the actions associated with them, either overtly or without alerting the individuals using these objects to the increased audit activity. The collaborative group may require summary audit reports on data transfers, storage, and retrievals to be published at a regular interval within the group. The audit information needs to be of adequate quality to meet the needs of the organisation, including the rigor required for forensic evidence in law. A key driving principle in a COA-related audit is transparency between partners.

(Forum 2008e)

Obviously, these services are definitely related to one another. These relations however are out of the scope of this thesis for now.

Attributes of the solution:

The fourth group consists of the attributes of the solution. They may assist in measuring whether preset objectives will be achieved, that could be accomplished by implementing the framework. The following attributes have been specified and may thus be chosen for using the framework:

Attributes of the Solution:
- Usability/Manageability
- Confidentiality
- Integrity
- Availability
- Efficiency/Performance
- Effectiveness
- Agility

Figure 45: Attributes of the Solution (part of the COA framework).

- **Usability/manageability**: Security measures are non-intrusive and are easily understood by the individual end-user.
- **Availability**: A collaboration's information should not be rendered unavailable either by mistake or by an adversary. This implies that any 'at rest' encryption keys are escrowed and that information is held in open-standard formats.
- **Efficiency/Performance**: Security measures do not greatly affect the latency, bandwidth, or total cost of data retrieval, storage, or transmission. This implies that collaborating partners must possess the means to rapidly access decryption keys for all data in their possession, for which they continue to have access privileges, allowing rapid data retrievals and offline malware scans.

- **Effectiveness**: The COA framework provides an effective approach to organizing and controlling secure data transport and storage among a wide range of existing and future corporate information systems.
- **Agility**: The COA framework takes into account the dimensions of timelines and flexibility. It enables development of business-driven enterprise architectures that are appropriately flexible and adaptable to facilitate changes in business operations with optimal speed and ease, but with minimal disruption.

(Forum 2008e)

The Fifth Group: Technologies:

As one can see at Figure 46 and Figure 36, a fifth group is displayed. However, it is not further discussed in (Forum 2008e), so we provide a free interpretation for now based on the drawings:

The technologies are divided into three groups:

- **End-point security / assurance:** this group focuses on establishing end-point security and assuring the right trust level for the devices in the transaction chain.
- **Secure communication** focuses on providing secure end-to-end communications between the collaborating entities. It is based on a set of secure protocols and covers wireless, mobile management, VoIP, Internet Filtering and Reporting and a set of Encapsulation and Encryption protocols.
- **Secure data** focuses on securing the information assets with several instruments based on Digital Rights Management mechanisms (which are often referred to as Enterprise Information Protection and Control).

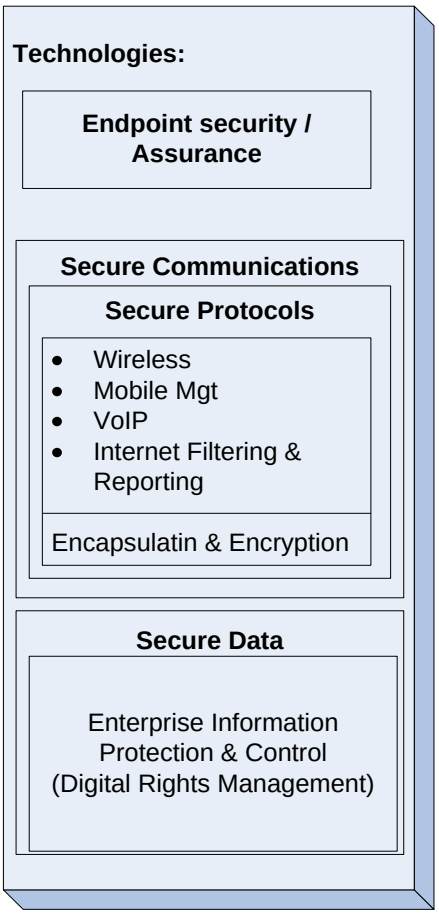


Figure 46: Technologies (fifth group of the COA framework).

Concluding:

We identify five groups:

- **Principles**, consist of requirements and constraints allowing one to collaborate safely if he follows these.
- **Processes,** consist of the PRIDE set, allowing one to use the SLATES elements in a collaboration.
- **Services,** are necessary to safeguard the collaboration processes.
- **Attributes,** can be used as goals to achieve with the framework, or as a set of measurements to see whether preset goals are achieved or not.
- **Technologies,** should be used to provide a safe end-point, secure end-to-end communications and secure information assets.

COA V2.0 UPDATES- part one:

The following relevant additions have been made in the COA 2.0 revision:

- **Structural changes**, in: "Position Paper – COA Framework", which can be summarized as follows:
  - *Enterprise relationship management refered to as called Enterprise Management:* in both this and "Enterprise Lifecycle Management" the Enterprise Relationship Management Enterprise is also refered to as Enterprise Lifecycle Management.
  - *Information asset management refered to as "Information Taxonomy and semantics":* Eventhough the process is renamed, it still covers the same aspects.

COA V2.0 UPDATES – part two:

- *Changes to trust management:* trust management now includes business impact levels, information classification, impact sensitivity categorization, control stratification and architecture segmentation model. See the paper and section 3.7.3 for more details.

The publications can be found at: https://www.opengroup.org/jericho/publications.htm

3.3.3. Framework Purpose in short

Now that we have seen the architect's overview we can try to partly address research question number two: "Why is it important?". For this, we must look at the purpose of the framework as we find it in the current documents and literature.
The purpose of the framework has been described in (Forum 2008e) from different perspectives:

- To operate securely in an environment of increasing information threats, where there is an ever-growing desire to interact without boundaries, irrespective of the location of the data or the number of collaborating parties.
- To respond to de-perimeterisation.
- To answer to JFC4-JFC8.
- To enable provision of IT systems that are secure in a global networked world, able to keep pace with the growing threats and the business need for faster and more flexible collaborative business arrangements. These range from outsourcing to joint ventures, from merger today to divestment tomorrow, all within a global working, global manufacturing and global procurement environment.

(Forum 2008e)

Summarised, it gives the following main purpose:
To provide companies an answer to de-perimeterisation and to enable them to collaborate securely by realising the Jericho Forum Commandments 4, 5, 6, 7, and 8.

3.3.4. How to adopt the framework in short

As visible in intermezzo 1, the framework can be adopted by integrating it into an architecture. The Jericho Forum has shown in (Forum 2008e) that one should adopt the COA framework elements into different information architectures. These information architectures can be Enterprise -, Network- or System information architectures.

3.3.5. Summary: The Collaboration Oriented Architecture Framework in short

Looking back at this paragraph, we can summarise it as follows:

- **The Collaboration Oriented Architecture** is the answer to de-perimeterisation and will allow organisations to collaborate securely by realising JFC 4 to 9.
- **COA framework consists of** five groups with different components. The groups consist of principles, processes, services, attributes and technologies.
- **Adoption can be realised by** implementing the components of the framework into information architectures of enterprises, networks or systems.

As noticed in chapter 1, the COA framework is yet unfinished. This automatically means that the findings in this paragraph are limited to the sources (e.g. the part of the framework) that

are finished. We will use the information retrieved here in order to research the main value and detail the five groups of components.

3.4.Main Value: SLATES, collaborative possibilities and security

3.4.1.Introduction

As the COA framework has been clarified a little previously, we now look at its main value. This allows us to answer research question number two ("Why is it important?"), which is done through different perspectives. We will look at it from: the Enterprise 2.0 checklist SLATES (section 3.4.2), the collaborative possibilities (section 3.4.3) and the value to de-perimeterisation and information security (section 3.4.4). We summarise our findings in section 3.4.5.

3.4.2.SLATES

As seen in section 3.3.2, SLATES is an Enterprise 2.0 checklist. Enterprise 2.0 would basically boil down to successfully using Web 2.0 applications inside an enterprise. There are multiple web 2.0 applications such as Blogs, mash-ups, online communities, social bookmarking, networking and Wikis.
McAfee has defined enterprise 2.0 as :

> *"The use of emergent social software platforms within companies, or between companies and their partners or customers"*
>
> (Hinchcliffe 2007)

That has been explained as:

> *"Social software enables people to rendezvous, connect or collaborate through computer-mediated communication and to form online communities. (Wikipedia's definition).*
> *Platforms are digital environments in which contributions and interactions are globally visible and persistent over time.*
>
> *Emergent means that the software is freeform, and that it contains mechanisms to let the patterns and structure inherent in people's interactions become visible over time.*
> *Freeform means that the software is most or all of the following:*
> - *Optional*
> - *Free of up-front workflow*
> - *Egalitarian, or indifferent to formal organisational identities*
> - *Accepting of many types of data*

> *Examples of Enterprise 2.0*
> - *DrKW's internal blogs and wikis*
> - *Rite Solutions' prediction markets*
> - *Enterprise tagging*
> - *R&D departments' use of Innocentive to find solutions to problems that have been stumping them.*
> - *MK Taxi's ability to connect mobile phone users in Tokyo directly to the driver of the cab closest to them, bypassing the dispatch center altogether.*
> - *Employee blogs like this one*

> *Not examples of Enterprise 2.0*
> - *Wikipedia, YouTube, Flickr, MySpace, etc. These are for individuals on the Web, not companies. Some companies use sites like YouTube for viral and stealth marketing, but let's explicitly put these activities outside our definition of Enterprise 2.0.*

- *Most corporate Intranets today. As discussed earlier, they're not emergent.*
- *Groupware and information portals. Again, these tools don't facilitate emergence, although this may be starting to change. Groupware and portals also seem to be less freeform than the Web 2.0 technologies now starting to penetrate the firewall.*
- *Email and 'classic' instant messaging, because transmissions aren't globally visible or persistent. Some messaging technologies do ensure that contributions are persistent"*

(McAfee 2006)

Hinchcliffe has defined enterprise 2.0 applications as:

> *"Social applications that are optional to use, free of unnecessary structure, highly egalitarian, and support many forms of data"*

(Hinchcliffe 2007)

Arguably, Enterprise 2.0 is about giving users the ability to get more involved in information creation, synthesising and processing all the way inside an organisation. This can be done by some of the web 2.0 applications, but also by completely different applications as in the example of MK Taxi.

Enterprise 2.0 allows the users to interact more freely and share the information inside the organisation that benefits most of the users. Others say it is more than only a set of tools: enterprise 2.0 is a mindset based on social information sharing inside and sometimes to the outside of the organisation.

The rise of these tools is less controllable by the management. America On-Line for instance could not control its own upcoming Mediawiki, neither in its creation nor in its growth. Making sure the information is secured is often more difficult.

Enterprise 2.0 has become a buzzword: managers like to use Enterprise 2.0 applications and tools for their users. However, how can one be sure they are using the right ones? McAfee introduced SLATES as a mnemonic to make it easy for everyone to remember what appeared to be the key aspects of these social platforms for the enterprise 2.0 domain.

(Farber 2006; McAfee 2006; Hinchcliffe 2007)

SLATES explained:

What exactly is SLATES? And what are the benefits of its elements? Understanding its origin, the SLATES-concept deserves a closer observation. As we follow (Forum 2008e), we understand that the Jericho Forum wants one to use the COA framework as an enabler to apply the concept of Enterprise 2.0 in a more secure way.

SLATES is defined as follows:

- **Search** is about the discoverability of information inside the organisation. Users can search for information in- or outside an organisation and use those search results online, by reusing the information for their own web 2.0 applications or other concepts. By enabling users to search within all information inside the enterprise, they have better insights of what happens and will be able to easily update their knowledge on the different topics that are interesting for the enterprise. It becomes much easier to spread information if it is indexed for all users. This allows more efficient reuse and a better return on investment of the published information.
- **Links** linking one or more sites to each other. This helps users to create links between different sites, pages or subjects in- and outside an enterprise. If someone has found information about a certain topic, that someone may use links to direct other users to that information. This allows users to sooner find the information. It also allows better reuse of the information that has been found. Enterprise content can become much more transparent by using URIs, because the tools become interconnected and related to each other.
- **Authorship** improves transparency to management and other users. It allows one to see whether a user is a consumer of information or a producer/synthesizer. One can look up the (nick)name of a co-worker and see if he or she published information. It can also multiply the amount of information, if this co-worker has written multiple blogs or wiki pages about a

certain topic. All pieces under the name of this co-worker as a search term lead the searching person to see whether the co-worker has published more interesting content on a subject.

- **Tags** allow users to organise the enterprise data by tagging sites or pages of an intra-/inter-/extranet site. It is a non-hierarchical keyword or term assigned to a piece of information. Tag tools allow users to find all the sites with similar tags allowing natural, organic, on-the-fly organisation of data from every point of view. This makes it easier for users to find information about topics by searching via tags throughout the different nets (intra-/extra-/internet).[98]
- **Extensions**: Extend knowledge by mining patterns and user activity. By monitoring actions of users, certain patterns are derived and others may be recommended to perform a certain action if part of a pattern is recognised. A good example is Amazon's recommendation system that recommends other items based on the users' behaviour. If multiple users bought a set of books, related to each other by the observed 'habitual' action, another user buying one or more items that follow the observed strain may be directed by the recommendation system to (buying) other books from that strain. The recommendation is thus based on multiple users having shown a pattern of buying a certain combination of items. The extension tools may help users to find their information faster and on-target. It also allows management to see the kind of information that has become a necessity to their workers. Understanding this, it allows them to manage the information in such a way their workers can access it as efficiently as possible.
- **Signals** allow users to see if any of their objects of interest have been updated. These objects may vary from RSS-feeds, to Web Slices et cetera, allowing users a more efficient information consumption.

(Farber 2006; Hinchcliffe 2007)

The Jericho Forum sees SLATES as key transformational elements that change the way organisations do business. When used in a collaborative relationship, it allows users of different enterprises to exchange information in a much more efficient way. This would also mean the information of the collaborative partner needs to be searchable, 'tagable' et cetera.
(Forum 2008e)

We can see that the Web 2.0 concepts are already delivering SLATES to the consumer, but we have not seen the enterprise 2.0 concepts do as much for the enterprises as web 2.0 has for the consumer (see Table 5).[99]

[1] There are many search engines and applications to enhance them.
[2] There are many search engines and applications to enhance them, yet many enterprises have not adopted the tools. Cross-organisational searches are often very hard or not available.
[3] There are many content management systems and other applications in the consumer market that allow users to execute links whenever they want to.

Comparison of application of SLATES in the consumer market and enterprise market		
SLATES	*Consumer market*	*Enterprise market*
Search	V[1]	V/X[2]
Links	V[3]	V/X[4]
Authorship	V/X[5]	V/X[6]
Tags	V[7]	V/X[8]
Extensions	V/X[9]	X[10]
Signals	V[11]	X/V[12]
Average application SLATES	V	X

Table 5: Comparison application SLATES in the consumer and the current enterprise market. (source: interview with A. Seccombe (April 2008), CSO Elli Lilly)

[98] See also http://en.wikipedia.org/wiki/Tag_(metadata) for more details, visited at 22-10-2008.

[99] At the other hand, one could say that, implementing the COA framework into the consumer World, would mean that it allows one to fully secure the Web 2.0 concepts.

[4] Most intranet systems allow users to paste links to other content as long as it is in line with company policy. However, most users are not allowed to simply create some content and put links in them.
[5] There are various ways to search for more items by a certain author. However, not all of them have been implemented as well as they could be.
[6] There are multiple tools, policies and regulations to establish the authorship of a certain object. However, there are many issues on this field, varying from ill using authorship tools to not maintaining the authorship of the objects themselves or the tooling. This becomes even worse during collaborative relationships: the authorship of documents from users from other companies is often harder to determine.
[7] There is a very large tagging community on the consumer net. Tag clouds and more are being created on a daily basis.
[8] There are tools for tagging inside the enterprise domain. However, not many companies have implemented anything like it. Although some have, it was without advertising to their own employees or in public.
[9] Some of the consumer market companies such as Amazon and social network sites such as Facebook and Hyves are currently working with the concept. However, there are still many organisations on the consumer market that ought to do more with it.
[10] Extensions inside the enterprise domain are used little to none inside the enterprise information domain.
[11] Signals are used often in the consumer market, varying from SMS alerts to e-mails, from widgets to browser plug-ins; all are build for alerting the user that his favourite content has been updated.
[12] There are multiple applications of signals in the enterprise domain. However, many organisations fail to have them implemented for objects in their enterprise information domain.

Reasons for the incapability of applying SLATES in an enterprise

There are multiple reasons for the lack of implementation and support for SLATES inside the enterprise of today. Many of them can be found in (Hinchcliffe 2007) and are not in scope of this thesis. However, some issues create the lack for support of SLATES that are witnessed by the author in different organisations:

- **The fear of losing control** over the information: if SLATES is applied to the enterprise information, the flow of information would not be as transparent anymore.
- **The lack of collaborative skills and knowledge** of employees and their management. Some are afraid their information will be reused without getting any reward for their efforts.
- **The lack of support from security and IT departments** to enable the interconnection between multiple information silos, departments et cetera to implement the tools that allow the enterprise to use SLATES.
- **The lack of supporting processes** necessary in order to guarantee certain continuity in the application of SLATES.

The value of the COA framework from the SLATES perspective

The COA framework can deliver great value from the SLATES perspective, in various ways:

- **Information protection and management**: The COA framework supplies multiple services and technologies that provide the capability to manage the information streams and monitor them. One will be in control of his own content and the reuse of it, even outside their own (cross-organisational) information domain.
- **Enabling collaboration**: The COA framework has many measures to enable safe collaboration between members of an organisation and between organisations themselves. It allows users to collaborate safely. By allowing safe collaboration and support for SLATES, both of the collaborative partners will gain the capability of accessing and easily finding the information inside their own domain and that of the other collaborative partner.
- **Enable trust management**: In order to maintain the collaborative relationships, trust management will be necessary. This should help the employees to trust one another and collaborate with them.
- **Enable secure interconnection**: The framework will allow organisations to interconnect for instance their departments and silos. This in turn allows for a much richer, broader and better-interconnected enterprise information domain, which makes the Enterprise 2.0 concepts flourish. It will even go further: as soon as two COAs interact, with the same means

of secure interconnection, they possess an improved interconnection between these parties, allowing for an even richer and broader cross-organisational information domain.

- **Provide supporting processes**: The COA framework delivers all of the necessary supporting processes in order to be capable of using Enterprise 2.0 tools, which in turn enable the use of SLATES inside an enterprise and across a collaborative relationship.

Summarizing:
The concepts of Enterprise 2.0 and SLATES will allow users to gain information far more efficiently. It allows companies to gain the maximum benefit in terms of information exchange during collaboration. It is already used in the consumer market as Web 2.0 and provides a high value to the masses inside the enterprises with Enterprise 2.0.
The COA framework improves value in terms of support and security to SLATES. By providing trust management, means of secure interconnection, information protection and management, and the necessary supporting processes, it is the answer to give the necessary support to enterprise 2.0 tools in both a cross-organisational collaborative relationship and an intra-organisational collaboration.

3.4.3. Collaborative possibilities

When looking at the value of the COA framework from the perspective of collaborative possibilities, one can find many valuable points:

- **Answer to SOA Collaboration:** As there has been a struggle to create a successful secure SOA collaboration platform, the COA framework may hold the answer. It allows SOA-based enterprises to create, establish, use and destroy a collaborative relationship fast and securely. (see sections 0 and 2.2.5 around the issues for SOA collaboration and the COA framework)
- **SaaS' trust management and security issues**: SaaS is currently having trust issues (as visible under section 2.3.3). These issues can be resolved by using the trust management the COA framework offers. It provides the customers and the providers in the SaaS market with dynamic collaborative relationships. The information protection mechanisms allow users to synthesise the information through different SaaS platforms and providers all together, without having to be afraid the information will be compromised or tampered with. This allows the SaaS-market to use their full collaborative potential and flourish even more (See also section 3.11.2).
- **Enabling fast dynamic relationships**: As relationships between companies take months to be built (see also section 2.6.8) and often do not have a good breakup procedure, it has become difficult to enable a dynamic relationship like the ones for the Jericho zone in the collaborative landscape (see also section 2.5.7 for the Jericho zone). The COA framework allows organisations to enter the Jericho Zone, build relationships extremely fast and break them up again without problems or trust-issues.
- **Collaborative relationships with a little negation to prisoners dilemma**: by using a good trust management system, Risk Management, protection services, audits and a clear set of processes for lifecycle management of (collaborative) partners, devices and enterprises, one can negate part of the Prisoners' Dilemma that we face in today's collaborative relationships. It is more transparent to both parties what consequences may arise and it will be easier to trust someone that an action, which has been agreed on, will be taken. (See also section 2.5.5 about the Prisoners' Dilemma)
- **Safeguarding information (in the collaborative relation)**: By using the services inside the COA framework, one can fully manage and protect one's information assets. Even if the documents spread out through collaborative relations and on the net, one still has the capability to protect that information and take full control over it.
- **Better manageability of relationships**: the relationships with other parties will be more manageable by use of the COA framework processes.

- **Enhancing (security) support for internet business models**: the current internet business models will be more secured by the services and processes of the COA framework (see also section 2.5.6 about the internet business models).

(Forum 2008e)

3.4.4. Value to de-perimeterisation and information security

The COA framework will give enterprises the answer to de-perimeterisation. It follows JFC4 to 8 and allows organisations to work according to these commandments[100]. The framework is not a techno-bullet, but it will help by applying technologies, processes and services to de-perimeterise securely and in a controllable way.
The framework also delivers value in terms of information security. The quality attributes of the framework itself show what the framework can deliver: better usability/manageability of the information security measures, better availability of the information itself, improved efficiency/performance of the security measures and higher effectiveness and agility for secure data transport and use.

3.4.5. Summary: the importance of the COA framework

Summarised, the value of the COA framework:

- **Enable enterprise 2.0:** The COA framework enables the secure usage of Enterprise 2.0 tools inside one's own information domain as well as in the cross-organisational information domain during a collaborative relationship. The COA framework delivers multiple advantages and valuable concepts such as better information protection and management, collaboration enabling processes, services, principles and technologies, better trust management, improved security of interconnection and the necessary supporting processes for the Enterprise 2.0 environment.
- **Provide an answer to current collaborative problems for SOA and SaaS**: the current issues around SOA Collaboration and around SaaS can be resolved by implementing and/or adopting the COA framework.
- **Enable fast dynamic better manageable relationships**: the COA framework enables fast dynamic relationships as occurring in the Jericho Zone of the collaborative landscape. The relationships will be more manageable as well by using the designated COA framework elements.
- **Enhance (data) security and management**: Data and its security can be managed more efficiently. Data both inside and outside an enterprise information domain is improved in manageability and protection.
- **Provide the answer to de-perimeterisation**: the COA framework delivers the answer to de-perimeterisation. It will allow companies to safely de-perimeterise.
- **Provide better information security**: Applying the COA framework to the information architectures, will create a better information security standard inside the organisation using these architectures.

The advantages found in this paragraph are only part of the list of advantages the COA framework could deliver to those who adopt it. As the COA framework is in a developmental phase, additional processes, services and concepts could be added to it, further improving its value.

[100] The commandments are also refered to as the "Jericho Principles". De-perimeterisation is also refered to as "Open Security".

3.5.COA principles

3.5.1.Introduction

As we have described the COA framework in a broader perspective (section 3.3.2), we will now try to detail it (see Figure 47).

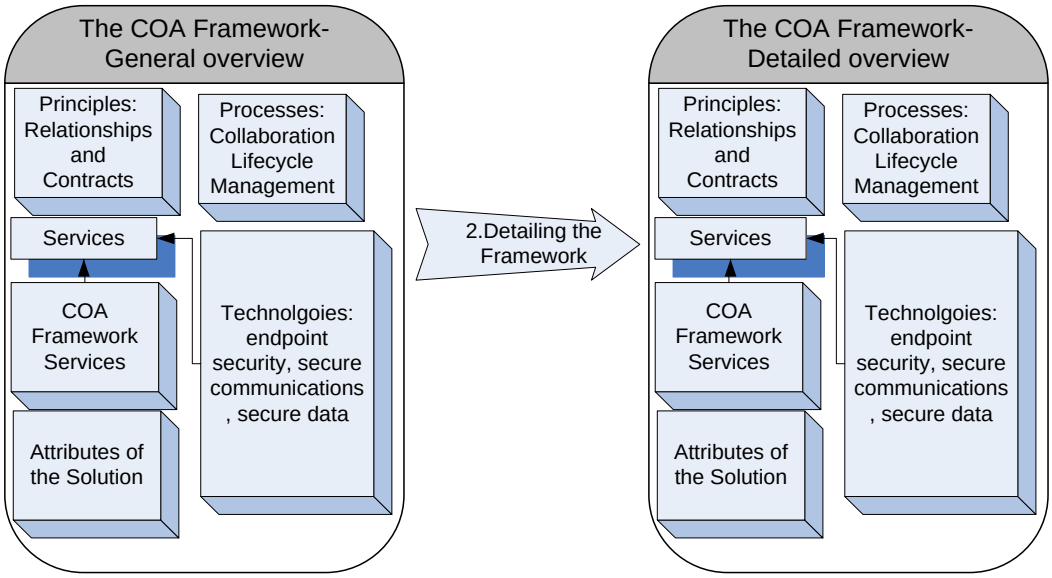


Figure 47: Overview of paragraph 3.5, 3.6, 3.7, 3.8 and 3.9.

The COA principles will be further elaborated on in section 3.5.2, which we will summarise in section 3.5.3. However, the elaboration is the first one since the introduction of (Forum 2008e) and cannot be seen as complete. More research has to be done based on this elaboration. This should be enough to start describing the first implementation of the COA framework though.

3.5.2.The principles explained

The principles have already been covered in section 3.3.2. We will now try to further elaborate on them by explaining what would be necessary to fulfil them, based on the chapter 2. All of the principles will be dealt with (see Figure 48).
See paragraph 3.10 for a mapping between the principles and the commandments.

Principles:
Relationships and Contracts:
- Participating parties
- Trust
- Assurance
- Risk
- Compliance
- Legal, regulatory, Contractual
- Privacy

Figure 48: Principles of the COA framework.

Participating Parties (know who- or what – you are transacting with):
This principle is about knowing with whom and with what a transaction is executed. Every party is responsible for corporate or individual entities that are well defined and whose activities are controlled by legal, economical, ethical, and technical means. What does this mean? What consequences does this principle have? The understanding itself is one of the basics in creating a situation of (mutual) trust. No one can trust someone without knowing who it is (see section 2.6.8 for more details around trust).
This principle implies:

- **The need for Identity Management**: in order to understand with whom one is transacting, Identity Management is a necessity. Authentication, verification and accounting are important to the principle, as well as the recommendations and items discussed in section

2.6.7. The Identity Management is already conveyed inside the framework as one of its services (see also sections 3.3.2 and 3.7.2 for more details on Identity Management inside the COA framework).

- **The need for End-point security:** In order to understand with what you are transacting at a device level, End-point security and device authentication are a necessity. In fact, End-point security is about raising the level of inherent trust in computing devices to a point where all meet the criteria of trust for that transaction. The recommendations and items discussed in section 2.6.9 are of importance to this principle. They can be found in the technologies (Mobile Management) and partly in the Device Lifecycle Management (see also sections 3.6.5 and 3.9.2).
- **The need for a trust management framework**: to be capable of seeing someone's state or trustworthiness, more than just the accounting data is of significance. It may be wise to use the reputation subsystems that need to be incorporated in a trust management framework as described in section 2.6.8. Trust Management is covered by the COA framework services. See section 3.7.3 for more details.[101]
- **Understanding collaborative relationships:** understanding with whom one is transacting in the transaction chain implies that one should understand the relationship itself. Not only the relationship matters, the kind of party, the other involved stakeholders, the influence of outside relationships to the transaction chain as a whole and as separate relationships and subjects, the bindings and roles the party can possess/perform, all influences the transaction. See section 2.5.4 for more details. A part of this understanding is covered in the Enterprise Relationship Management process, see section 3.6.6 for more details. Most of the complexity the understanding brings will have to be covered by strategic thinking by the enterprise's management, collaborating within the chain.

Trust (agree to the level of trust/confidence you will be transacting at):

The second principle is about understanding the level of trust/confidence one will be transacting at. This means the collaborating parties can agree/define appropriate (known) degrees of confidence in the components in a transaction chain, including the environment in which the components are operating. If someone knows at what level he is transacting in terms of confidence, he simultaneously determines the level of trust required for the transaction throughout the transaction chain.

This principle is a logical requirement: if one wants a trustworthy transaction, he should first understand when it is trustworthy. If highly classified information is exchanged, a high trust level is an absolute necessity. Understanding this, allows parties to dynamically choose the required level of trust, depending on the context and the value of the information being exchanged.

Looking back at chapter 2, we can say that this principle implies:

- **The need for a trust management framework:** as with the previous principle, there will be a need for the use of a trust management framework, which allows entities to create a trust relationship and monitor the trust level of that relationship. (See section 2.6.8) Trust management will be covered by the COA framework services. See section 3.7.3 for more details.
- **The need for data classification:** in order to know what level of confidence is necessary, one should also classify the information assets related to the transaction and communicate the outcomes of the classification throughout the chain. See section 2.6.6 for information on classification. The COA framework resolves it with its services. It has a service for information classification, see section 3.7.3 for more details.

[101] It is also optional to include connectors to the trust management framework from the Enterprise Relationship Management processes. See section 3.6.6 for more details.

- **Understanding privacy matters:** The privacy of the data influences the level of confidence that is necessary to deal with it. See section 2.6.6 for information on data privacy. The services of the COA framework cover this. There is a service for information classification (and the relating privacy matters), see section 3.7.3 for more details.
- **The need for Risk Management and -taxonomy:** Risk Management and –taxonomy cover the level of confidence for the complete chain to communicate the risks and thus obtain the necessary trust level. These are covered by the Risk Management process in the processes group. See section 3.6.3 for more details.
- **The need for inter- organisational information policy management**: The information access policies should be exchanged in order to help other parties in the transaction chain understand the required level of privacy. Different organisations may have different policies and different classifications derived from their policies. That is why the information access policies should be exchanged in order to help every member of the transaction chain understand what level of confidence needs to be used. See section 2.6.5 on policy management. Policy management will be covered by one of the COA framework services, see section 3.7.4 for more details.
- **Understanding collaborative relationships:** like the previous principle, understanding the influence of all stakeholders, roles and relations is crucial, to see if the trustworthiness still holds. Understanding the roles of the participating entities inside the network helps assessing the trustworthiness that can be provided in the transaction. However, part of the influence of the stakeholders, roles and relations can be negated by the use of the trust management framework. See section 2.5.4 and 2.6.8 for more details. Part of this will be covered by the Enterprise Relationship Management process, see section 3.6.6 for more details.

Assurance (verify that the agreed level of confidence pertains):
The third principle follows the second principle: after understanding the level of trust/confidence the transaction will take place at, the participating parties must agree on the level of trust/confidence that is used in the transaction itself. This means each party should agree with the required level prior to the transaction and then cling to that agreement. Assurance in doing so can be provided by contracts or other binding artefacts that can be used to safeguard the intellectual property. By agreeing on the kind of security that is applied, the way in which it is applied and what will be done (or not) with the retrieved information, additional assurance can be given.
One should understand it is important to first ascertain the level of trust that is necessary from the intra-organisational perspective and only then agree to the level of trust that is required by inter-organisational demands. It may be the case those two levels are not the same. If so, additional communication and/or different types of assurance become necessary in order to achieve at least the required level of intra-organisational trust and still be a good, efficient and trustworthy player in the transaction chain.
Looking back at chapter 2, we can say that this principle implies:

- **The need for a trust management framework:** Again, a trust management framework is necessary, this time for monitoring the trust level and getting assurance the trust level pertains among the complete chain. The artefacts used for assurance can also reside within the broker of the trust management framework. See section 2.6.8 for more details. Trust management is covered by the COA framework services. See section 3.7.3 for more details.
- **The need for audit and accounting:** in order to provide more assurance than only a contract, auditing and accounting will have to take place for each party in the transaction chain. This is covered by the audit service and the lifecycle management processes (see section 3.7.6 and paragraph 3.6 for more details).
- **The use of inter-organisational governance:** to integrate the audit and accounting processes, inter-organisational governance is necessary (see paragraph 2.4 for details on (IT) governance). The need for governance will be partly fulfilled by the current audit service of the COA framework (see section 3.7.6 for more details)

- **The need for inter- organisational information policy management**: as we have seen with the previous principle, the policies have to be used to see the required level of confidence. The inter-organisational information policy management is necessary again to enforce the communicated level of confidence. See section 2.6.5 on policy management. Policy management is covered by one of the COA framework services, see section 3.7.4 for more details.
- **Understanding collaborative relationships**: one will have to understand and stick to his role in the collaborative chain to see whether he can demand certain data from the collaborating parties. Even though the trust management framework can negate some of the influence of the roles of the party inside the transaction chain, one's current position still influences the possibilities to gain assurance from other parties. An organisation in the commander role can still gain more assurance data with less effort than an organisation in a subordinate role. See section 2.5.4 and 2.6.8 for more details. Part of this will be covered by the Enterprise Relationship Management process, see section 3.6.6 for more details.

Risk:

The fourth principle is about understanding the risk within and surrounding the transaction. The collaborating parties can assess any proposed transaction based on the communicated levels of trust with factors germane to the transaction: identity, confidentiality, integrity, availability, location, environment (space it is being used in), data-sensitivity, transaction value, time et cetera.

The risk assessment is a valuable step to setting the required level of trust (second principle).

This principle implies:

- **The need for a trust management framework:** the trust management framework can be used to obtain information about the components that have to be assessed. Reputational and other information can be provided by the framework (or the broker for that matter) in order to do a risk assessment. See section 2.6.8 for information about the trust management framework. Trust management is covered by the COA framework services. See section 3.7.3 for more details.
- **The need for audit and accounting:** in order to provide data for the risk assessment alongside the information that has been provided by the trust management framework, auditing and accounting are needed for each party of the transaction chain. This is covered by the audit service and the lifecycle management processes (see section 3.7.6 and paragraph 3.6 for more details).
- **Understanding collaborative relationships:** one has to understand his role in the collaborative chain to understand the additional risks that may be part of the transaction. For example, an organisation in the role of the commander may take a position in the transaction chain that lowers the risks for himself in the chain, while a subordinate may be put in a position in which he faces more risks than the commander, or then initially agreed on. See section 2.5.4 for more details around the subject of collaborative roles and relationships. A part of the assessment is covered by the Enterprise Relationship Management process, see section 3.6.6 for more details. Most of the complexity this understanding brings will have to be covered by strategic thinking of the management of the enterprises that collaborate inside the chain.
- **The need for data classification:** in order to understand the level of risk that comes with the transaction, one should also classify the information assets that are related to the transaction and communicate the outcomes of the classification throughout the chain. See section 2.6.6 for information on classification. The COA framework resolves this by its services. It has a service for information classification, see section 3.7.3 for more details.
- **Understanding privacy matters:** The amount of personal identifiable data that is necessary for the transaction will positively influence the level of risk to the transaction. See section 2.6.6 for information on data privacy. The COA framework resolves this by its services. It has

a service for information classification (and the relating privacy matters), see section 3.7.3 for more details.

- **The need for Risk Management and -taxonomy:** In order to assess the risk and publish the outcome, one needs Risk Management and taxonomy to define the outcomes. These are covered by the Risk Management process in the processes group. See section 3.6.3 for more details.
- **Need for Identity Management and end-point security**: It is necessary to understand the who and what involved in the transaction, to assess the risk involved with the transaction. This means there is a need for Identity Management (see section 2.6.7 around the matter) and end-point security (see section 2.6.9 for more details). The first is conveyed in the COA framework services (see also sections 3.3.2 and 3.7.2 for more details on Identity Management inside the COA framework) and the latter partly in the technologies (Mobile management) and partly in the Device Lifecycle Management (see also sections 3.6.5 and 3.9.2).
- **The need for assurance, compliance and governance:** in order to be capable of assessing the risk, one has to check for historical records and future promises of assurance around the collaborative relationship and the future transaction that can be provided partially by the COA framework processes (see paragraph 3.6). The same goes for compliancy information. Furthermore, proper governance processes and controls have to be checked to see if there are any additional risks in the transaction chain. (see paragraph 2.4 for details on (IT) governance). The need for governance is partly fulfilled by the current audit service of the COA framework (see section 3.7.6 for more details)

Compliance:

The fifth principle is about compliancy: one should comply to the rules and regulations of the security inside the collaborative group. Collaborating parties must agree to periodic inspections and security audits. The results of these inspections and audits are published within the collaborative group. Non-compliant parties may be sanctioned or expelled.

Looking back at chapter 2, we can say that this principle implies the following:

- **The need for (IT) audit, accounting and (IT) governance:** to be able to audit each other and publish the results inside the collaborative group, a proper set of governance and accounting processes is necessary. The set provides proper information exchange and adequate quality of the security in all of the enterprise business and IT processes (see also paragraph 2.4 and sections 2.6.10 and 2.6.7 on these subjects). This is covered by the audit service and the lifecycle management processes (see section 3.7.6 and paragraph 3.6 for more details).
- **The need for a trust management framework:** in order to publish the audit results inside the collaborative group, one can use the repository of the trust management framework. The trust management framework can also use its broker as a place to keep the security agreements. See section 2.6.8 for information about the trust management framework. Trust management is covered by the COA framework services. See section 3.7.3 for more details.
- **The need for secure technologies for data protection:** to be compliant with the security agreements, one needs secure technologies, inherently secure protocols and standards to secure the data in all cases. However the technologies that will be required are compliancy artefact depending: the group will have to make its own agreements on what security has to be used and what standards have to be adhered to. See the following sections around security: 0, 2.6.9 and 2.6.4. Most of the necessary security is covered by the COA Technologies (see paragraph 3.9 for more details) and the audits are incorporated in the COA Services (see section 3.7.6 for more details).
- **Understanding privacy matters:** Personal Identifiable information that is checked during audit, will need to be handled with care. See section 2.6.6 for information on data privacy. The COA framework resolves this by its services. It has a service for information classification (and the relating privacy matters), see section 3.7.3 for more details.

- **The need for policies:** In order to enforce compliancy, one needs a set of policies to for instance information access management and business processes. See section Policy Management for more details. The policy management is covered by the respective COA framework service, see section 3.7.4 for more details.
- **Understanding collaborative relationships:** one has to understand his role in the collaborative chain to see whether he or she can really get the access necessary to a complete security audit. Even though the trust management framework can negate some of the influence of the roles of the party inside the transaction chain, the position one has highly influences the possibilities on the field of auditing. It is easier for an organisation in the commander role to set up and execute a full audit of an organisation in a subordinate role than vice versa. See section 2.5.4 and 2.6.8 for more details. Part of this will be covered by the Enterprise Relationship Management process, see section 3.6.6 for more details.
- **The need for end-point security:** In order to maintain security on all devices, End-point security is necessary. The recommendations and items discussed in section 2.6.9 are significant to this principle. It can be found in the technologies (Mobile management) and partly in the Device Lifecycle Management (see also sections 3.6.5 and 3.9.2).

Legal/Regulatory/Contractual:
The sixth principle is an addition to the fifth. Besides complying to security rules and regulations inside the group, one should also comply to other agreements, rules and regulations. This may vary from contractual requirements in- or outside the group to legal requirements (by law or others) and regulatory requirements. All members of the group should be able to resolve conflicts that may arise between them, through effective verification and enforcement mechanisms. If companies undertaking a transaction reside in different countries, the laws of those countries should be followed by all parties. The same goes for the regulatory means for companies of different sectors. The principle shows in (Forum 2008e) that compliance to local, legal and regulatory requirements alone is unlikely to be good enough to meet all business requirements. Additionally, one should look for compliancy to other concepts such as good entrepreneurship, corporate governance.
This implies:

- **The need for proper governance, accounting and audit:** to maintain the different agreements, rules and regulations, (e.g. establish “effective verification”) one will need proper inter-organisational and intra-organisational governance (see paragraph 2.4 for some of the aspects of governance) and proper audits and accounting (see sections 2.6.10 and 2.6.7). This is covered by the audit service and the lifecycle management processes (see section 3.7.6 and paragraph 3.6 for more details).
- **The need for awareness and understanding of local and sector dependent legal and regulatory means:** One needs to understand the local and sector-dependant legal and regulatory means, to comply to them. This is partly addressed by the enterprise relationship management process (see section 3.6.6) and the trust management service (see section 3.7.3).
- **The need for a penalty supporting system:** To “effectively enforce”, one needs a supporting penalty system and hold the accounting information. This is created based on the trust management framework and thus covered by the trust management services (see section 3.7.3).
- **The need for a trust management framework:** The trust management framework is necessary in order to exchange the (local/sector-dependant) legal and regulatory means and support the penalty system (See section 2.6.8 for information on the trust management framework). Trust management is covered by the COA framework services. See section 3.7.3 for more details.
- **Understanding privacy matters:** As soon as checkups are done for compliancy, personal identifiable information will be used. It follows that one has to understand the privacy matters and deal with them properly. See section 2.6.6 for information on data privacy. The

COA framework resolves this by a service for information classification (and the relating privacy matters), see section 3.7.3 for more details.

- **The need for policies:** policy exchange can increase the efficiency of "effective enforcement mechanisms". See section 2.6.5 for more details on policies. The policy management is covered by the respective COA framework service, see section 3.7.4 for more details.
- **Understanding collaborative relationships:** one has to understand his role in the collaborative chain to see if he can really get the access that is necessary for a complete audit to all legal, regulatory and contractual compliancy. This can be problematic for creating the right contractual means. Even though the trust management framework can negate some of the influence of the roles of the parties inside the transaction chain, one's position influences the possibilities on the field of auditing. It is easier for an organisation in a commander role to set up and execute a full audit of an organisation in a subordinate role than vice versa. See section 2.5.4 and 2.6.8 for more details on the roles and relationships inside a network of collaborating organisations. Part of this is covered by the Enterprise Relationship Management process, see section 3.6.6 for more details.

Privacy:
The seventh principle is privacy. Privacy is a particularly significant requirement for the collaborating parties to meet. Increasingly, privacy is being defined in legislative safeguards, which is the consequence of a widespread belief in privacy as a fundamental human right. At its root are customers, suppliers, and employees, expecting organisations to use information about an individual ethically so that it is not divulged if it is reasonably considered "private".
Looking back at chapter 2, we can say that this principle implies the following:

- **The need for proper information management:** to handle privacy with care, one needs proper information management and protection (see section 2.6.6). This is dealt with by the COA framework information asset management service (see section 3.7.5) and the information lifecycle processes (see section 3.6.4).
- **The need for proper classification:** To detect the Personal Identifiable Information that could create privacy issues, proper information classification is necessary (see section 2.6.6). This will be dealt with by the COA framework Trust Management and Classification services (see section 3.7.3).

Benefits and Obligations:
The eighth principle is not entirely finished. In (Forum 2008e) it handles benefits and obligations. A repository of obligations and requirements such as contractual obligations, service level agreements (SLA), customer expectations, corporate policy, and norms of good corporate citizenship are requirements that need to be aligned and implemented. Detailing the stated requirements means mostly following the lines of the sixth principle. No further elaboration will be made on this principle yet.

Links between the principles:
Many of the principles are linked to each other, see Table 6, Table 7 and Table 8 on the next pages.

3.5.3. Summary: the COA Principles

There are seven COA framework Principles that are highly interrelated and consist of requirements and constraints such as participating parties, trust, assurance, risk, compliance, legal/regulatory/contractual, privacy, benefits and obligations. They are significant to ensure a safe (set of) collaborative relationship(s) between (multiple) companies that are trustworthy.

They also help keeping those relationships trustworthy on a longer term. If one does not follow these principles he becomes less trustworthy and unsafe for collaboration.
Most of the principles imply processes and/or services already incorporated in the framework.
Normally, these principles only apply to events that happen outside the enterprise domain. As the boundary between the outside and the inside fades away because of de-perimeterisation however, they will apply to the inside of the domain as well.
One should notice the current principles are still in a developmental stage (as the Benefits and obligation- principle is not finished).

Principles	*A. Participating parties*	*B. trust*	*C. Assurance*
1. Participating parties	-	See A.2	See A.3
2. trust	To trust someone/ something, he/she will have to understand who/ what it is/they are.	-	Assurance needs to be given based on the level of trust that is necessary in the transaction chain.
3. Assurance	One will have to know who the other is before he/she can ask for any form of assurance. Knowing who/what something is can already provide a certain level of assurance.	Trust is partly depending on the accounting information that has been given for the assurance.	-
4. Risk	In order to assess the risk, one will have to know with who and what he is transacting.	The level of communicated and assessed trust influences the level of assessed risk. If a high level of trust is required by third parties, then the level of risk could be higher as well.	The level of risk is depending on the level of assurance that can be provided by the components of the transaction chain.
5. Compliance	Compliancy information can help to see whether the identities are correct.	None directly related to the principles.	Compliancy information of the past can be used as an addition to the artefacts for assurance.
6. Legal/ regulatory/ contractual	The components for the transaction chain are bound to legal, regulatory and contractual means.	The trust level on which one will be transacting can be influenced/enforced by legal/ regulatory/ contractual means	Legal/regulatory/ contractual artefacts can act as artefacts of assurance.
7. Privacy	PII will be necessary for understanding who and what you are transacting with. This automatically means that privacy arises.	Privacy matters will influence the level of trust/confidence that is necessary for the transaction.	Personal identifiable information will have to be used to identify the components in the assurance artefacts, so privacy issues will arise.
8. Benefits and obligations	One will have to understand with who and what he is transacting in order to fulfil the other	The level of trust that is necessary can be influenced by the exact implementation of the obligations stated here.	The different artefacts that will arise because of the obligations, can, depending on their type, be used as artefacts for

	obligations.		assurance.

Table 6: relationships between the COA Principles, part one.

Principles	*D. Risk*	*E. Compliance*	*F. Legal / regulatory / contractual*
1. Participating parties	See A.4	In order to check for compliance with regulations, one will have to understand with whom and what he is transacting.	See A.6
2. trust	The level of risk is based on the trustworthiness of the other parties. If a party is less trustworthy or communicates a lower level of required trust, then the risk of the transaction will increase.	None directly related to the principles.	See B.6
3. Assurance	The necessary assurance from other parties is based on the level of risk and the required trust level found in the risk assessment.	See C.5	See C.6
4. Risk	-	The compliancy artefacts from the past and the promised artefacts for the future can be used for risk assessment.	The legal / regulatory/ contractual artefacts from the past and the promised artefacts for the future can be used for risk assessment.
5. Compliance	See E.4	-	The two principles are additions to each other.
6. Legal/ regulatory/ contractual	See F.4	See F.5	-
7. Privacy	The privacy issues surrounding or within the transaction will influence the results of the risk assessment. The more PII is found, the higher the risk.	There will be audits on the field of information security and respect to privacy within the collaborative group.	Privacy issues will be covered by regulatory, contractual and legal agreements.
8. Benefits and obligations	The artefacts that create certain obligations will influence the outcome of the risk assessment process.	The alignment of several obligatory components such as corporate policies will affect the requirements that will be set up for security audits inside the collaborative group.	The legal / regulatory / contractual artefacts can help to align the obligatory means such as policies and the idea of good corporate citizenship.

Table 7: Relationships between the COA Principles, part two.

Principles	G. Privacy	H. Benefits and obligations
1. Participating parties	See A.7	See A.8
2. trust	See B.7	See B.8
3. Assurance	See C.7	See C.8
4. Risk	See D.7	See D.8
5. Compliance	See E.7	See E.8
6. Legal/ regulatory/ contractual	See F.7	Resolving conflicts due to breaking any of the rules that are embedded in the legal / regulatory / contractual artefacts can be resolved by the common understanding of good corporate citizenship and other obligatory means.
7. Privacy	-	As soon as PII will be necessary for effectively creating and using obligatory artefacts, privacy issues will arise.
8. Benefits and obligations	See H.7	-

Table 8: Relationships between the COA Principles, part three.

3.6.COA processes

3.6.1.Introduction

Processes:
Collaboration Lifecycle Management:
Management processes for:
- People
- Risk
- Information
- Devices
- Enterprise Relationships

Figure 49: COA framework processes.

After having described the COA framework (section 3.3.2) and detailed the COA principles, a detail of the COA processes is given, based on what we know from chapter 2 and the previous paragraphs. See also Figure 47 in section 3.5.1.
The processes can be seen as separate sets of different processes that allow management of the assets mentioned in Figure 49. The sets will be shortly described based on available knowledge. Following, a set of requirements and recommendations will be given per process set. The following sets are described in this paragraph:

- The People Lifecycle Management processes in section 3.6.2.
- The Risk Management processes in section 3.6.3.
- The Information Lifecycle Management processes in section 3.6.4.
- The Device Lifecycle Management processes in section 3.6.5.
- The Enterprise Relationship Management processes in section 3.6.6.

Section 3.6.7 summarises the observed items. The processes together form the word PRIDE (Person, Risk, Information, Device and Enterprise). The processes allow one to implement SLATES and the benefits of enterprise 2.0 in an enterprise. The value and definitions of SLATES are discussed in sections 3.3.2, 0 and 3.4.2, and will not be repeated here. However, we will look per process set what their addition could be to SLATES.

As seen in (Forum 2008a; Forum 2008b), most of the processes are related to each other. Most of those relations remain out of the scope of this thesis. The relation needing attention is:

An individual (Identity Management) can only be allowed access to certain information if the asset itself allows it (Information Lifecycle Management) based on the type of device and the security status of that device (Device Lifecycle Management), the enterprise he belongs to (Enterprise Relationship Management) and the total risk involved (Risk Management).

3.6.2. People Lifecycle Management

Introduction:
The People Lifecycle Management processes aim at managing individuals in their process of joining, operating and departing the collaboration. All personas included in the collaborative endeavour should be managed. Many things need to be considered such as identities, personas, capabilities, reputation, and the potential impact of each of the individuals. (Forum 2008e)

There is currently a limited amount of information available about the People Lifecycle Management. Only (Forum 2008e) acknowledges its existence, but no other papers give information on the subject. That is why most information has to be inferred from the support to SLATES, the COA framework Principles, Services and the findings in chapter 2.[102]
Following, a short description of the processes is given. From there on recommendations and requirements for those processes will be added.

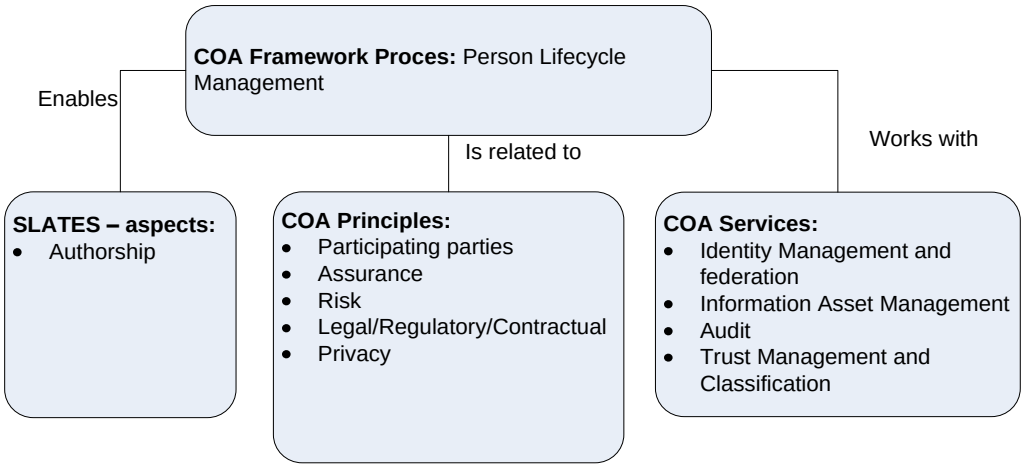


Figure 50: Relations between People Lifecycle Management, SLATES, Principles and Services.

Relation to SLATES:
In order to be capable of establishing authorship (SLATES) of information assets, one has to manage the author entities and their respective identity. This asks for Identity Lifecycle Management and ultimately for a more complete set of processes: People Lifecycle Management.

Relation to the COA Principles:
There are multiple relations with the COA Principles, this process set connects to or supports the following principles:

- **Participating parties**: Because the 'who' in a transaction must be known, a set of processes is necessary that allow identification. Since every party is responsible for corporate or

[102] Important related sections: 2.6.7 around Identity Management.

individual entities and their activities, one has to manage different aspects: the complete Identity Lifecycle, one's reputation, his capabilities (in order to make sure that he will not have to do anything he cannot that will lower his reputation) et cetera. This means the People Lifecycle Management processes should take care of almost all information aspects around a person.

- **Assurance** and **Risk**: The reputational state of an individual is important to both having some assurance in a non-contractual form and the assessment of risk for a certain transaction. Assessing the risk however requires more than only the reputational state. In order to assess the risk in an individual, one has to understand his position in the transaction chain and his capabilities as well.
- **Legal/Regulatory/Contractual:** The person and/or identity-based contracts and legal and regulatory objects applicable to a certain person, which is related to a collaborative relationship, should be clear to the managing entity. This means all of the information that comes from legal, regulatory and contractual items should be managed by these processes as well.
- **Privacy:** as loads of PII will be processed by these processes, the principle of Privacy should be taken in mind as well.

Relation to the services defined in the COA framework:
The People Lifecycle Management processes can be related to the following COA framework Services:

- **Identity Management and Federation:** In order to manage the lifecycle of a person, one has to manage his Identity Lifecycle as well. This must be done by the Identity Management and Federation service (see section 3.7.2 for more details).
- **Information Asset Management:** the results of the information classification and its resources will be used in the Information Asset Management service.
- **Audit**: the personas have to be audited to see how they work and if their actions are in line with the policies.
- **Trust Management and Classification**: to connect to the trust broker and exchange information that is either produced by or necessary for these processes (see section 2.6.8 for more details). This service is also necessary to detect PII and understand the kind of information assets used by a person during his lifecycle in the collaborative environment. The information needs to be classified. The services also have to be used to classify all information that is created by these processes.

A short description of the process:
The processes are based on the idea of Identity Management in a broader context. See for the background information the following sections: 2.6.4, 2.6.6, 2.6.7, 2.6.8 and 3.3.2.

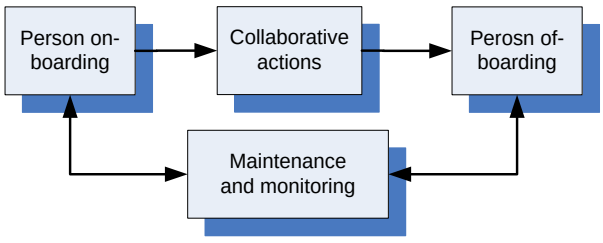


Figure 51: People Lifecycle Management.

Name: People Lifecycle Management (PLM)	
Description: PLM is about managing the complete lifecycle of the identity of a persona during a collaborative relationship with that persona. This relationship can vary from being an employee of the company to being an external stakeholder. A person will first on board and then execute a set of actions during a certain time span. Meanwhile his actions should be monitored and his identity, reputation, capabilities, competencies and impact to the organisation should be managed. Al of those will change because of the actions of the individual. PLM consists of multiple processes: Person on-boarding, Person maintenance, monitoring, person off-boarding.	
Name process: Person On-	*Description:* handles all of the required aspects for enrolling a person to an organisation as a member or to a collaborative relationship as an external

boarding	member. Internal members will get complete account provisioning (which differs based on the identity model that is used, as soon as user centric ids are possible, that will be the process), internal reputation and capability records. All important legal, regulatory and contractual means will be registered as well in the trust broker repositories. The history of the member and impact to the organisation will be assessed and registered as well. The Person roll in will be the starting process which allows users to be provisioned with the necessary equipment in terms of hard- , software, ID-s et cetera. All of the data will be registered at the trust broker as well, with the approval of the user, which manages the PII part of that information in collaboration with the trust broker and the organisation he is working with. The organisation can also acquire historical information, with the user consent that is considering its PII(see sections 2.6.6, 2.6.7, 2.6.8 and the COA privacy principle). External members of the collaborative relationship will be capable of using their ID inside the organisation during the collaborative effort. Reputation, history and capability reports are requested and provided with users consent. The external member will also be provided with the extra materials that he needs for his task in this collaborative relationship
	Interacts with following services: Identity Management (identity provisioning or re-acquiring for internals, identity acquirement for externals), Classification and Trust management (to check the trustworthiness and history of both internal and external members). See sections 3.7.2 and 3.7.3 for more details.
Name process: Person Monitoring	*Description:* all of the attributes and actions of a persona that are related to the collaboration will be monitored. Information requests, updates and creation will be logged, as well as transactions. Actions that do not change the trustworthiness or the reputation of the user will not be available to collaborative authorities, only to the trust broker and may only be accessed by users consent.
	Interacts with following services: Classification and Trust management (to update and monitor logs). Audit, for auditing the logs and the user actions. See sections 3.7.3 and 3.7.6 for more details.
Name process: Person Maintenance	*Description:* this process will keep record of the actions and monitoring interventions from the Person monitoring. Changes and additions to the capabilities, reputation, repositories for legal/regulatory/contractual artefacts and impact to the organisation will be recorded and further dealt with by this process. Maintenance actions can be seen by the subject itself and the PII involved will be managed by the subject in collaboration with the trust management framework and the authorities.
	Interacts with following services: Classification and Trust management (to update and monitor logs). Audit, for auditing the logs and procedures of this process, Identity Management for updating the identity and related objects and attributes. See sections 3.7.3, 3.7.6 and 3.7.2 for more details.
Name process: Person Off - boarding	*Description:* this process will allow organisations to de-provision the identity or de-activate certain parts (depending on the identity type). The company equipment will be returned as well. Furthermore, the PII-rights will be fully returned to the subject that will leave the organisation or the collaborative relationship. Only the basic public records and the legislative artefacts will remain with the organisation, the rest will only reside with the trust broker that will have a relation with the subject. The information can be accessed by anyone with consent of the subject.
	Interacts with following services: Trust management and Classification (to deal with the trust broker). Audit, for auditing the logs and procedures of this process, Identity Management for de-provisioning the identity and related

	objects and attributes. See sections 3.7.3, 3.7.6 and 3.7.2 for more details.

Table 9: Summary of People Lifecycle Management processes.

Requirements:

The following requirements can be defined for the People Lifecycle Management, based on sections 2.6.6 (Privacy), 2.6.7 (Identity Management), 2.6.10 (Audit). 2.6.8 (Trust, -management and – brokers) and the Jericho Forum Commandments:

- Requirements derived from the Identity Management and related fields[103]:
  - The processes should support identity federation and, as soon as the technology is available, user centric identity.
  - The processes should support the currently available identity and transport protocols.
  - The processes should be reliable, auditable and easily manageable.
  - The processes should allow additional information to be stored by either the Identity Management service or in one's own meta-information repositories.
  - The processes should take the laws of identity into account.
  - The processes should be location- and protocol-insensitive: one should be capable to execute them from any location and with any set of protocols that covers all other requirements.
  - The requirements of authorisation and authentication as seen in section 2.6.7 should be taken in mind as they are part of the Identity Management and the people lifecycle processes.
- Requirements derived from Audit[104]:
  - The processes should be completely auditable.
  - The processes should contain all necessary metadata and controls, allowing the auditor to understand the necessity of another type of planning and scope for auditing.
- Requirements derived from privacy:
  - Respect to privacy: the subject to the PII should stay in control of the information.
  - The processes need to support different privacy platforms.
  - The processes should save all the meta-information considering any PII-related actions and those of the processes.
  - All PII should be protected and co-managed by the subjects in order to protect their privacy.
  - The subjects should be noticed of the PII and gain full access to, as well as the decisions on, that data.
- Requirements derived from trust and – management[105]:
  - The processes should support oncoming trust management standards and protocols, and their respective repositories for reputation, digital identity, contractual information et cetera.
  - The processes should be capable of accessing legal frameworks in order to use the country-/legal- specific procedures for the entire People Lifecycle Management.
  - All persons should be registered at the trust broker.
- Requirements derived from the Jericho Forum Commandments:
  - The scope and level of protection should be specific and appropriate to the asset at risk, meaning there should be a variable set of protection measures taken in mind per process, identity, reputation et cetera. This also means the assets in these processes should be protected per asset.
  - Security mechanisms for these processes must be pervasive, simple, scalable and easy to manage.

[103] See for the other requirements derived from / on Identity Management section 3.7.2.

[104] See for the other requirements derived from / on Audit section 3.7.6.

[105] See section 3.7.3 for the other requirements derived from / on Trust and -management.

- The processes should use open and secure protocols and standards that may be used for multiple types of devices.
- The level of trust of a person should be transparent throughout the complete set of processes.
- The trust management framework will have to be used in order to let authentication, authorisation and accountability be exchangeable outside one's locus/area of control.
- The data, and access to that data, should be properly protected throughout the complete set of processes.

Recommendations:
We define the following recommendations for People Lifecycle Management based on chapter 2, paragraph 3.3 and this section:

- **Secure communications and message protection between the processes:** The messages and the communications will have to be properly secured. Depending on the content of the messages, different types of protection will have to be taken into account. See section sections 0, 2.6.4 and 2.6.6 for more details.
- **Co-development with COA framework services:** to make sure the processes are executable, one should develop them alongside their most important services such as the Identity Management and the Trust Management service. This requires parallel roadmaps to the services, streamlining such a development.
- **Privacy management system:** a management system should be designed with a PII broker to protect the PII and to let the users be in control, as seen in section 2.6.6.
- **Use of OASIS standards until PII broker:** In order to manage the PII one should first try to use already available standards such as WS-privacy in order to manage the information, until the PII broker is realised. See sections 0 and 2.6.6 for more details.
- **Service based implementation:** the process should be implemented based on services, allowing one to take the full benefits of the available SOA.
- **The creation of an open standard:** the processes should be detailed and standardised in an open and inherently secure standard, which is protocol and model independent and allows multiple vendors to come up with solutions that will be interoperable.
- **The creation of a process paper:** the Jericho Forum should publish a process paper that supports the development of the standard.
- **Additional governance:** an additional governance framework should be designed for auditing these processes and providing the right governance tools for their management.
- **Further research of the processes:** all of these processes must be further researched to provide the needed solutions. One should take the current SOA security concepts in mind as found in section 0 and 2.2.4.

See for more recommendations the related sections in chapter 2 and the related COA Services in paragraph 3.7.

COA V2.0 UPDATES:

As this section is an impression to the COA V2.0 updates as they have not been finished yet:

- **Upcomming additions to the COA V2.0 Revision**, in "Person Lifecycle Management", which can be summarized as follows:
  - *Renaming the process to "Person Lifecycle Management"*. The process will be renamed in de Rev.2.0 of the framework.
  - *The need for a master source of data.* There is a need for a master source of user information, which might be an authoritative source for systems wishing to federate.
  - *The need for a strong personal ID*. There is a need for a strong personal ID, more information about this will be revealed later when the position paper is ready.

The publication will be published at: https://www.opengroup.org/jericho/publications.htm

3.6.3. Risk Management

Introduction:
The Risk Management processes consist of processes, methods and approaches that identify, classify, and manage the information risks involved in collaborations. One has to take sources of information risk into account for the assessment. This may vary from liabilities by a lesser reputation of an organisation or an individual, to technological sources of risk such as faulty or absent implementation of certain security measures. (Forum 2008e)

An increasing stream of information is becoming available to understanding Risk Management. The Jericho Forum and the Open Group are working on multiple documents to further detail Risk Management such as (Forum 2008e; Fox 2008; Jones 2008) and the oncoming Risk Management process paper. All of them are still under development and the sources (Fox 2008; Jones 2008) have been seen as of the scope of this thesis simply due to time restraints. We will try to discuss this process set in short, based on SLATES, the COA Principles, and the findings of chapter 2. Furthermore, we refer to (Fox 2008; Jones 2008) for more details and to encourage the reader to work through them as soon as they are completely available. The process-description is an attempt based on limited data. Additional research in this field is required, even after this section.

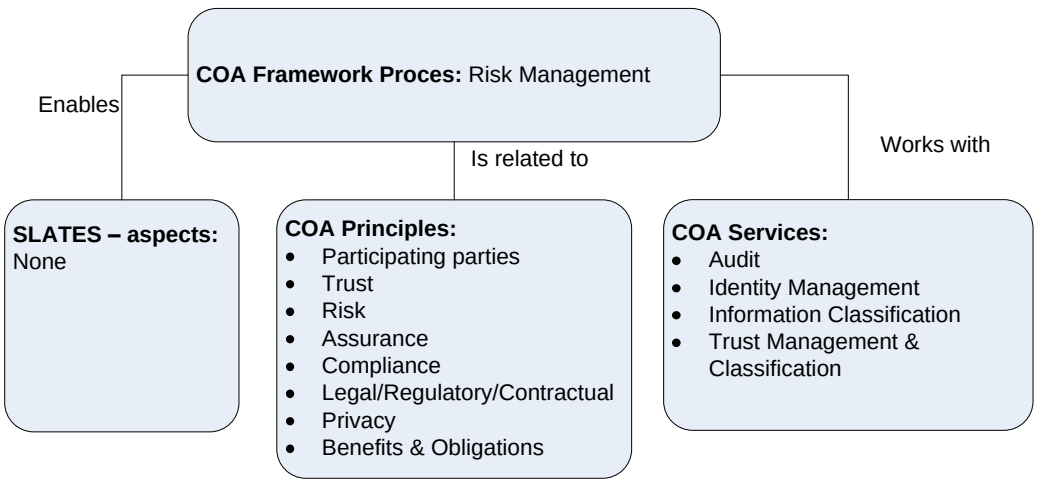


Figure 52: Relations between Risk Management, SLATES, Principles and Services.

Relation to SLATES:
The only link that can be found, is that data for risk assessment is easier obtained if all parties in the collaborative relationship for the transaction have implemented SLATES in such a way that it is usable in an inter-organisational fashion.

Relation to the COA Principles:
Multiple related to the COA Principles, this process set connects to or supports the following principles:

- **Participating parties:** in order to be capable of a proper risk assessment, one must know with whom and what he is dealing. This automatically links Risk Management to other processes such as Person-, Device- and Enterprise Lifecycle Management.
- **Trust:** the required level of trust can be derived from the risk found in a risk assessment. The necessary level of trust can be lowered if one will use proper Risk Management to lower the risks.
- **Risk:** this principle is directly implemented by the Risk Management processes.
- **Assurance**, **Compliance** and **Legal/Regulatory/Contractual:** all of these have a direct impact on the Risk Management processes and vice versa. The current artefacts created from these principles can be used as input for the risk assessment and as assets to manage risk by reducing risk with these artefacts.
- **Privacy:** as loads of PII will be processed, the principle of Privacy should be taken in mind as well.
- **Benefits and Obligations**: The artefacts and processes that derive from this principle can be used as input for the risk assessment and as assets to manage risk by reducing risk with these artefacts.

Relation to the services defined in the COA framework:
The Risk Management processes can be related to the following COA framework Services:

- **Audit:** audits of all kind of controls and processes (both business and IT) and is necessary in order to do a proper risk assessment and proper Risk Management.
- **Identity Management:** the identities of all devices and subjects within the collaborative context have to be managed and used in the process of risk assessment and Risk Management.
- **Trust Management and Classification:** to connect to the trust broker and exchange the information that is necessary for the risk assessment and Risk Management. That same information can also be used to check the trust value of an asset. Furthermore, one has to know the risks to every information asset. That is why information classification is an important service to this process.

A short description of the process:
The processes descriptions are loosely based on the description we found in the beginning of this section, the results can be used as an indication or as an elaboration.

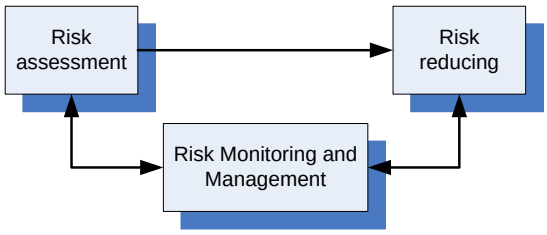


Figure 53: Risk Management (simplified).

Name: Risk Management (RM)	
Description: RM is about assessing and managing the risk that can come with transactions and the exposure of information assets. It checks for the risk that a valuable asset can become and for all vulnerabilities. It will Furthermore, either execute or propose certain risk reducing or security increasing measures.	
Name process: Environmental Threat	*Description:* This process will assess the threats from the current environment. This can vary from the reputation of a person or organisation that is currently in contact with the organisation to environmental threats and

Assessment	hazards (nature, et cetera)
	Interacts with following services: Identity Management, Federation and Reputation (section 3.7.2), Trust Management and Classification (section3.7.3) and Audit (section 3.7.6).
Name process: Information Risk Assessment	*Description:* This process will assess the information security risk that comes with the information assets and the combination of these assets.
	Interacts with following services: Trust Management and Classification (section3.7.3).
Name process: Transactiona risk assessment	*Description:* this process checks for all of the risks inside the complete transaction chain, with all the parties, individuals, their respective identities and reputations, the information assets involved and more taken in mind.
	Interacts with following services: Identity Management, Federation and Reputation (section 3.7.2), Trust Management and Classification (section3.7.3) and Audit (section 3.7.6)
Name process: Risk Monitoring	*Description:* this process is the monitoring process for all of the risks that have been identified by the other processes.
	Interacts with following services: Identity Management, Federation and Reputation (section 3.7.2), Trust Management and Classification (section3.7.3), Policy Management (section3.7.4) and Audit (section 3.7.6).
Name process: Risk Reducer	*Description:* this is the processes that will use the findings of the risk monitoring either to reduce the risk by itself or advise what actions have to be taken to reduce or counteract the risk.
	Interacts with following services: implementation depending, further research required.

Table 10: Indication of Risk Management processes.

Requirements:

Due to the lack of a good set of sources, no requirements will be defined.

Recommendations:

The Risk Management processes and the ideas behind the processes have to be studied further. The Jericho Forum should release a process paper that is (partly) based on the sources named in the introduction of this section.

COA V2.0 UPDATES-part one:

As this section is more an impression then a complete outline of risk management, the additional position papers in COA V2.0 will be quite helpful. The V2.0 release includes the "COA Position Paper Risk Lifecycle Management", which can be summarised as follows:

- **Movement from architectures to applications**. The risk analysis will move from architectures to applications, making the information risks more numerous and more severe due to the amount of elements and the less defence-in-depth.
- **Believes of the Jericho Forum:**
  - *Organisations need to manage risk in a way that is systematic* and closely relates to the organisation's business environment and security architecture.
  - *Organisations need to express and manage information risks* in the same way as any other risk. In particular, they should use methods that are, or can be made to be, quantitative.
  - *Extensive tool support will be required* by all but the smallest organisations to allow information risks to be managed properly.

COA V2.0 UPDATES –part two:

- **Relation between business context, security risks and security architecture**. These three influence each another:
  - The business context provides economic support for the security architecture, while it gains security and a structure from the security architecture.
  - The security architecture is justified by security risks, while the access and the probabilities defined in the security architecture are source of the security risks.
  - The security risks have a certain impact on the business context, while they themselves are input to system acceptance.
- **A need for standards**. There is a need for standards for:
  - The development and application of effective tools.
  - Evaluation and acceptance of risk, both within and between organizations.

  One will need the ISO27001/2, the Risk Taxonomy, FAIR and other additional standards. See the paper for more details.

The publication can be found at: https://www.opengroup.org/jericho/publications.htm

3.6.4. Information Lifecycle Management

Introduction:
The Information Lifecycle Management processes consist of all the processes that effectively and efficiently manage the creation, reading, update and deletion of information assets in a collaboration. These processes include audit, monitoring and information protection activities. (Forum 2008e)

There is still a limited amount of information available around the Information Lifecycle Management processes. Only (Forum 2008e) acknowledges existence, but no other papers provide information on the subject. That is why most of the information will have to be inferred from the support to SLATES, the COA framework Principles, Services and the findings in chapter 2.
Following, a short description of the processes will be given and then recommendations and requirements for those processes are added.

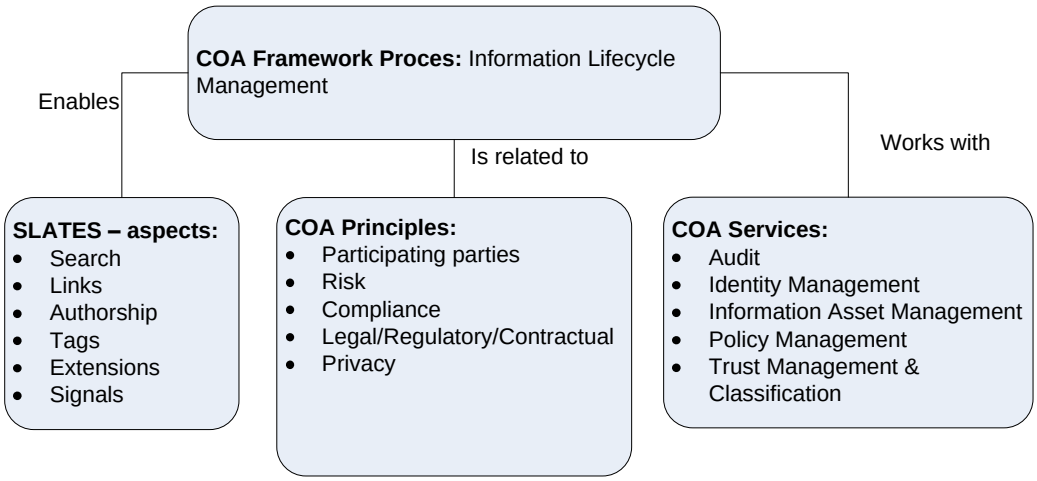


Figure 54: Relations between Information Lifecycle Management, SLATES, Principles and Services.

Relation to SLATES:

There are multiple relations to SLATES. One could argue these processes will manage all of the important ones for information and thus (partly) all of the aspects of SLATES. Proper management of the information makes for a better indexation, making it searchable. It also allows one to implement better signals and extension mechanisms based on the meta-information created by these processes. Authorship can be controlled and checked by using proper information construction and editing processes. The processes could enable tagging.

Relation to the COA Principles:
There are multiple relations with the COA Principles; this process set connects to or supports the following principles:

- **Participating parties:** the processes are connected to this principle as follows: in order to take full control over the information processes, one will have to know who is (or wants to) reading/updating/deleting what.
- **Risk:** The processes directly support the principles on risk. Understanding what exactly happens with what information, allows one to see the kind of risk that is involved in the transaction.
- **Compliance** and **Legal/Regulatory/Contractual:** The processes support these principles by providing and managing (meta-)information necessary for these principles. They also support compliancy and other regulations by information protection functions.
- **Privacy:** as loads of PII will be processed, the principle of Privacy should be taken in mind as well.

Relation to the services defined in the COA framework:
The Information Lifecycle Management processes can be related to the following COA framework Services:

- **Audit:** this service will be necessary for auditing the information protection activities.
- **Identity Management:** to understand who is reading/updating/ deleting what, Identity Management will be a necessity.
- **Information Asset Management:** this is one of the main services for the Information Lifecycle Management processes. Most of the information management will be done by this service (see section 3.7.5 for more details).
- **Policy Management:** to manage the information, one has to use proper information access policies (see also section 2.6.5 around that subject). That is why the policy management service is an important service for these processes as well.
- **Trust management and classification:** to be capable of exchanging the information necessary for this process or produced by this process with the trust broker, and to classify all of the information assets in order to manage them according to their classification.

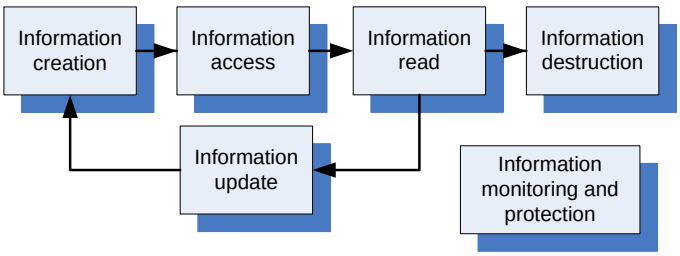


Figure 55: Information Lifecycle Management (simplified).

A short description of the process:
The processes descriptions are based on the findings of this section, section 3.3.2 and background sections 2.6.6, 2.6.5 and 2.6.8. Each of the steps named in (Forum 2008e) will be seen as a separate process:

Name: Information Lifecycle Management (ILM)
Description: ILM focuses on managing the complete information lifecycle of the information assets. It covers the creation, the access, the reading, the updating, the deleting and the safeguarding of the information. One should realise that the processes that have been identified as a process, consider still a set of processes in different that differ in implementation: there is no standard for exchanging all of the information: one could use a

XML message for instance or a PDF document.	
Name process: Information Creation	*Description:* This process covers the creation of the information. This means that the information itself is being defined in a certain container (e.g. XML message, PDF document et cetera) or that new information is added to the container alongside the existing information. From there on the new information will be classified in its context (e.g. other information, author, organisational context). The classification results will be used to select the right information protection mechanisms and information access policies and implement them into the container. The implementation process will have to take the legal, regulatory and contractual rules in mind.
	Interacts with following services: Identity Management, Federation and Reputation (section 3.7.2) for managing the authorship of the new information, Trust Management and Classification (section 3.7.3) for classifying the asset and connecting the process to the trust management broker, Policy Management (section 3.7.4) for inserting information access policies into the information container based on the information found during the classification process, Meta/Information Management (section 3.7.5) for managing the information asset and apply execute most of the process as a service and last: Audit (section 3.7.6) in order to audit the controls and this process. The information will also be registered at the trust broker and the information monitoring process in order to monitor the information.
Name process: Information Monitoring	*Description:* This process will monitor all of the information that has been created and pushed forward by the information creation process. It will record each (request to) access, update and destruction in detail and check if the information access policies are respected. It also monitors the flow of the information: to other containers, locations (e.g. physical and change of legal domains), users, et cetera.
	Interacts with following services: Meta/Information Management (section 3.7.5) for checking the status of the information asset, Audit (section 3.7.6) for auditing the controls in this process and Policy Management (section 3.7.4) for checking the status of the implemented information access policies.
Name process: Information Safeguarding	*Description:* the information should always be safeguarded in a way that is appropriate to the asset at risk. This means that this process should check if the information still needs to be safeguarded the way it was. It will execute a reclassification based on the asset and the context of the asset and if new security measures should be applied. If the reclassification shows that the security can be lowered, then measures could be taken away from the information asset. However, if the reclassification shows that there is an indication to take more measures, then more and/or stricter security measures and information access policies should be applied to the information asset in all of its current forms.
	Interacts with following services: Identity Management, Federation and Reputation (section 3.7.2) as part of the classification process, Trust Management and Classification (section 3.7.3) for classifying the asset and connecting the process to the trust management broker, Policy Management (section 3.7.4) for inserting information access policies into the information container based on the information found during the classification process, Meta/Information Management (section 3.7.5) for managing the information asset and apply execute most of the process as a service and Audit (section 3.7.6) in order to audit the controls and this process.
Name process: Information Spreading	*Description:* this process is responsible for spreading the information. Whenever one wants to send, copy or move the information, then this process will record these activities and either disallow or allow the actions depending on the identity and end-point security status. Report to the

	information monitoring process. It will also disallow unauthorised copies, movement and provide some control of data in the wild. In all cases of activation of this process, the monitoring process should be notified of the new actions (authorised or not).
	Interacts with the following services: Identity Management, Federation and Reputation (section 3.7.2) for authenticating the user and see if he is authorised. Meta/Information Management (section 3.7.5) for managing the information asset and apply execute most of the process as a service and Audit (section 3.7.6) in order to audit the controls and this process.
Name process: Information Access	*Description:* this process is responsible for the access to the information asset container (e.g. e-mail, web site, PDF-document, XML message). This process will differ depending on the security measures that have been taken to protect the information asset. One could think of different security approaches XACML (see 0) or DRM (see 2.6.6) or no security measures at all. It could also take the end-point security status in mind as well. In all cases of protection, the monitoring process should be notified of the new actions (authorised or not).
	Interacts with the following services: Identity Management, Federation and Reputation (section 3.7.2) for checking the identity, reputation and rights and the authentication, authorisation and identity related accounting processes. Policy Management (section 3.7.4) for checking and using the information access policies. Meta/Information Management (section 3.7.5) for managing the information asset and applying/to execute most of the process as a service and Audit (section 3.7.6) in order to audit the controls and this process. In all cases of protection, the monitoring process should be notified of the new actions (authorised or not).
Name process: Information Reading	*Description:* This process is responsible for ensuring that one can only read the information inside the information container that he is allowed to read. This will allow for partial blanking out. This process will differ depending on the security measures that have been taken to protect the information asset. One could think of different security approaches XACML (see 0) or DRM (see 2.6.6) or no security measures at all. It will also take the end-point security status in mind as well. In all cases of protection, the monitoring process should be notified of the new actions (authorised or not).
	Interacts with the following services: Identity Management, Federation and Reputation (section 3.7.2) for checking the identity, reputation and rights and the authentication, authorisation and identity related accounting processes. Policy Management (section 3.7.4) for checking and using the information access policies. Meta/Information Management (section 3.7.5) for managing the information asset and applying/to execute most of the process as a service and Audit (section 3.7.6) in order to audit the controls and this process.
Name process: Information Updating	*Description:* This process allows one to update the information if he is authorised to do so. It should also notify all other (users of the) copies of the information asset that a change has been applied. This process will differ depending on the security measures that have been taken to protect the information asset. One could think of different security approaches XACML (see 0) or DRM (see 2.6.6) or no security measures at all. It could also take the end/point security status in mind as well. In all cases of protection, the monitoring process should be notified of the new actions (authorised or not).
	Interacts with the following services: Identity Management, Federation and Reputation (section 3.7.2) for checking the identity, reputation and rights and the authentication, authorisation and identity related accounting processes. Policy Management (section 3.7.4) for checking and using the information

	access policies. Meta/Information Management (section 3.7.5) for managing the information asset and applying/to execute most of the process as a service and Audit (section 3.7.6) in order to audit the controls and this process.
Name process: Information Destruction	*Description:* This process is responsible for destroying the information asset (and all of its copies). The asset or assets can only be destroyed by someone who is authorised to do so. This process will differ depending on the security measures that have been taken to protect the information asset. One could think of different security approaches XACML (see 0) or DRM (see 2.6.6) or no security measures at all. It could also take the end-point security status in mind as well. In all cases of protection, the monitoring process should be notified of the new actions (authorised or not).
	Interacts with the following services: Identity Management, Federation and Reputation (section 3.7.2) for checking the identity, reputation and rights and the authentication, authorisation and identity related accounting processes. Policy Management (section 3.7.4) for checking and using the information access policies. Meta/Information Management (section 3.7.5) for managing the information asset and applying/to execute most of the process as a service and Audit (section 3.7.6) in order to audit the controls and this process.

Table 11: Processes description of Information Lifecycle Management.

Requirements:

The following requirements can be defined for the Information Lifecycle Management processes, based on sections 2.6.6 (Privacy), 2.6.5 (Policy management), 2.6.6 (Data classification, Protection and Privacy), 2.6.7 (Identity Management), 2.6.8 (Trust, -management and -brokers), 2.6.9 (End-point Security), 2.6.10 and 2.4 (Audit) and on the Jericho Forum Commandments:

- Requirements derived from Privacy: The same requirements as for the People Lifecycle Management processes (section 3.6.2) are needed. Additional requirements:
  - The information production and update processes should be executed by the subjects if only on PII.
  - The information needs to be destroyed by the organisation if no longer necessary.
- Requirements derived from Policy management[106]:
  - The information access policies should support multiple governance patterns such as automated control, workflow-based control, accountability and time-limited permissions.
  - The language should separate information access policy administration and policy enforcement.
- Requirements derived from Data classification[107]:
  - The processes should allow multiple types of classifiers and classification services.
  - The processes should handle all metadata as data.
  - The processes should allow reclassification and temporal classification on any given moment.
- Requirements derived from Data protection[108]:
  - The data protection measures should be in line with the legal requirements of the collaborative environment.
  - The process needs to be capable of handling multiple kinds of data protection measures, standards and protocols.

[106] See for the other requirements derived from / on Policy management section 3.7.4.

[107] See for the other requirements derived from / on Data classifcation section 3.7.3.

[108] See for the other requirements derived from / on Data protection section 3.7.5.

  - The information reading process needs to have the capability of blanking out parts.
- Requirements derived from Identity Management and related fields[109]: the same requirements as for People Lifecycle Management hold for Information Lifecycle Management.
- Requirements derived from Trust, -management and –brokers[110]: the same requirements as for People Lifecycle Management hold for Information Lifecycle Management.
- Requirements derived from End-point security[111]:
  - The processes should be capable of inspecting multiple end-points in multiple zones where the asset and its copies are located.
  - The processes should only allow spreading of information to devices where the necessary elements of the COA framework are active.
  - The processes should be protocol-independent for End-point Security.
- Requirements derived from Audit[112]: the same requirements as for People Lifecycle Management hold for Information Lifecycle Management.
- Requirements derived from the Jericho Forum Commandments:
  - The processes should always protect the information according to the value of the asset (JFC1).
  - The processes should be pervasive, simple and scalable over an unlimited amount of resources, devices and assets (JFC2).
  - The processes should be easy to manage (JFC2).
  - The processes should use open and inherently secure protocols (JFC3).
  - The primary information asset protection should happen at the level of the information containers and the metadata surrounding the information asset inside that container by using DRM or XACML (JFC9).
  - The processes need to support the trust broker for handling the permissions, keys, privileges et cetera (JFC10).
  - The processes should always ensure the data is appropriately secured when stored, in transit and when in use (JFC11).

Recommendations:

We define the following recommendations for Information Lifecycle Management based on chapter 2, paragraph 3.3 and this section:

- **Secure communications and message protection between the processes:** The messages and communications have to be properly secured. Depending on the content of the message, different types of protection have to be taken in mind. See section sections 0, 2.6.4 and 2.6.6 for more details.
- **Privacy management system:** a management system should be designed to protect the PII and let the users be in control, as seen in section 2.6.6.
- **Use of OASIS standards until PII broker:** to manage the PII, one should first try to use already available standards such as WS-privacy in order to manage the information, until the PII broker is realised. See sections 0 and 2.6.6 for more details.
- **Service based implementation:** the process should be implemented based on services, allowing one to take the full benefits of the available SOA.

[109] See for the other requirements derived from / on Identity Management section 3.7.2.

[110] See section 3.7.3 for the other requirements surrounding Trust and -management.

[111] As the end-point security will be added to the Identity Management service, see section 3.7.2 for all of the other requirements.

[112] See for the other requirements derived from / on Audit section 3.7.6.

- **Reading and access process should be separately implemented:** to allow partial blanking out (see also section 2.6.6) of documents, the access and reading processes must be separately implemented.
- **Process development together with services:** the process should be developed alongside its most important services (data classification and information asset management) in order to ensure it will be executable. This requires parallel roadmaps to the services for a streamlined development.
- **The usage of risk taxonomy and traffic light protocol:** There should be an easy way of communicating risk of the assets and the classification level of the data. Both the risk taxonomy and traffic light protocol add value here. See also section 2.6.6.
- **The creation of an open standard:** the processes should be detailed and standardised in an open and inherently secure standard, which is protocol and model independent and allows multiple vendors to come up with interoperable solutions.
- **Additional governance:** an additional governance framework should be designed for auditing the processes and providing the right governance tools for their management.
- **The creation of a process paper:** the Jericho Forum should release a process paper that supports the development of the standard.
- **Further research of the processes:** all of these processes must be further researched to develop solutions to them. One should take the current SOA security concepts in mind as found in section 0 and 2.2.4.

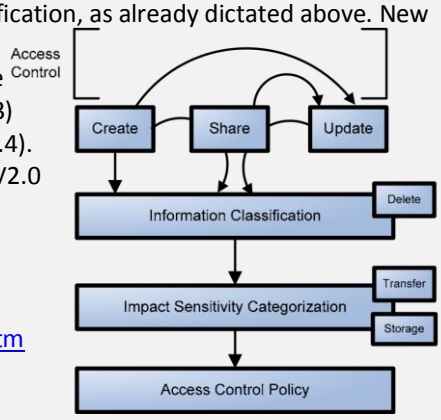
See for more recommendations the related sections in chapter 2 and the related COA Services in paragraph 3.7.

COA V2.0 UPDATES:

The following relevant additions have been made in the COA 2.0 revision:

- **A new process model** in "COA Paper Information Lifecycle Management", which can be summarized as follows. There is a new process model which could be seen as an extension of the process model advised in section 3.6.4. As an addition it takes the impact sensitivity categorization and access control requirement definition/modification process in account.
    - *Creation:* The creator of the information should check whether it requires access control and if so, apply the complete process.
    - *Storage and information sharing:* the security measures taken must reflect the information classification and impact sensitivity categorization.
    - Update and delete: the update must consider the information classification and impact sensitivity categorization while updating the information. This also counts for deletion of the information.

  The process model considers information classification, as already dictated above. New to the model is the Impact sensitivity categorization, which is further described at the V2.0 update of trust management (section 3.7.3) and at the update of access control (section 3.7.4). Access Control policies will be discussed in the V2.0 update of Policy Management (section 3.7.4). See the Position Paper for more details.



The publications can be found at:
https://www.opengroup.org/jericho/publications.htm

3.6.5. Device Lifecycle Management

Introduction:
The Device Lifecycle Management processes will be introducing devices, identifying and maintaining device trust levels, interconnections, and removing devices involved in collaborations. The processes could take care of all devices in the collaborative environment, but they aim at those devices that could be an "end-point" in any transaction chain (see section 2.6.9 for more details). Device Lifecycle Management will also cover Software Lifecycle Management (Forum 2008e; Forum 2008a)

The processes look like server management, but the server management processes will be more formalised and out of the scope of this thesis for now.
The description, requirements and recommendations of these processes is based on (Forum 2008a; Forum 2008e), the COA Principles, SLATES and the knowledge found in chapter 2.
Software Lifecycle Management is partly covered by these processes.

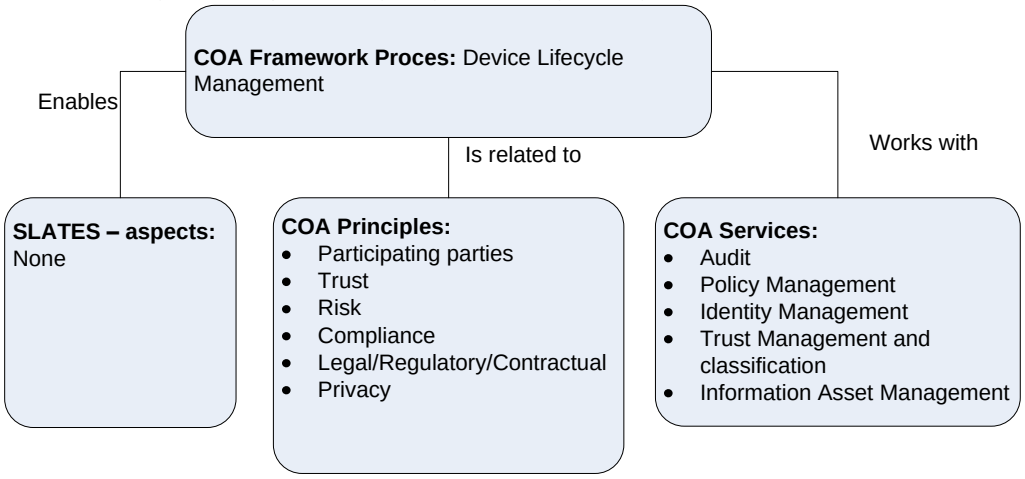


Figure 56: Relations between Device Lifecyle Management, SLATES, Principles and Services.

Relation to SLATES:
There is no direct relation to SLATES. The software management processes on the devices indirectly allow a complete support of all SLATES aspects, but there is no direct connection between SLATES and Device Lifecycle Management processes.

Relation to the COA Principles:
There are multiple relations to the COA Principles; this process set connects to or supports the following:

- **Participating parties:** to manage all information streams, one has to understand who and what is creating, editing, updating and/or reading the information with which software and on what device. One has to know with who and what he is dealing considering these information processes.
- **Trust:** in order to be capable or defining the necessary trust level, one will have to understand what kind of information with what the trust state can be of the devices that are involved within the transaction.
- **Risk:** these processes will create loads of metadata that can be used for risk assessment. Risk Management itself will be an important part of the processes in terms of managing the trust level of the devices.
- **Compliance:** compliancy to security standards will be a very important principle for these processes.
- **Legal/Regulatory/Contractual:** the artefacts that come from this principle will influence the processes even more. Agreements and laws around soft- and hardware will have a major influence on these processes.

- **Privacy:** as loads of PII will be processed by these processes, the principle of Privacy should be taken in mind as well.

Relation to the services defined in the COA framework
The Device Lifecycle Management processes can be related to the following COA framework Services:

- **Audit:** the controls and other means that will create a certain required trust level of a device will have to be audited. The processes themselves and the controls in those processes will have to be audited as well.
- **Policy management:** in order to manage the trust state of devices and software one will need proper policy management (and enforcement).
- **Identity Management:** this will also be applicable to devices in order to manage them completely. One will have to identify and manage the devices by their identity.
- **Trust Management and classification:** in order to record the state of the device and the software into the trust broker repository. It will also be used to classify all the information that comes along in the Device and Software Lifecycle.
- **Information Asset Management:** in order to manage all the information that comes along in the Device and Software Lifecycle.

A short description of the process:

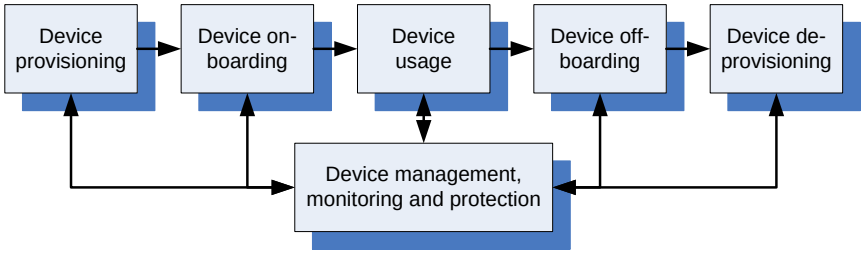


Figure 57: Device lifecycle management (simplified).

The Device Lifecycle Management process is based on the findings of the following sections: 2.2.4, 2.6.5, 2.6.7, 2.6.8, 2.6.9. 2.6.10 and 3.3.2.

<table>
<tr><td colspan="2">Name: Device Lifecycle Management</td></tr>
<tr><td colspan="2">Description: This set of processes takes care of the complete lifecycle of the devices that can be an end-point in a transaction chain. It Furthermore, takes care of the complete software lifecycle for the software on the devices.[113]
The lifecycle consists of the following processes: the device lifecycle consists of device provisioning, on-boarding, management and monitoring, off-boarding and additional processes such as device remediation.</td></tr>
<tr><td rowspan="2">Name process:
Device Provisioning</td><td>Description: This is the starting process for someone who will need new hardware to be allowed to do the required actions in his new role in the collaborative environment. A device will be provisioned to him and the other processes will be planned as well (device on-boarding until remediation and software provisioning until de-provisioning). All information of this process and the planning process will be saved in both the trust brokers repository as the enterprise information system under the ID of the device (which will be created in this process), related to the user ID.</td></tr>
<tr><td>Interacts with following services: Identity Management, Federation and</td></tr>
</table>

[113] It does assume that most of the software is in the cloud.

	Reputation (section 3.7.2) will be used for the Identity Lifecycle of the device. Trust Management and Classification (section 3.7.3) for allowing access to the trust broker. Policy Management (section 3.7.4) for managing all the policies around this process and on the device itself. Audit (section 3.7.6) to audit the controls and this process.
Name process: Device On-boarding	*Description:* When a device is provisioned or when another identity has the right devices, yet did not had any access with that device to the enterprise domain, then this processes will be executed. The Identity from the device should be added to the domain repository and registered as active in the domain in the trust broker repository. It should Furthermore, retrieve all the policies, agents and other software that is necessary for the domain. This will trigger the software provisioning process and the device monitoring process. The device will also be assessed for suitability of interaction based on amount of memory, applications, connectivity et cetera. This data will be used for optional additional software provisioning, allowances in the Information Lifecycle Management process and optional negotiation about the mutually agreeable method of interaction of the device with its environment.
	Interacts with following services: Identity Management, Federation and Reputation (section 3.7.2) will be used for the Identity Lifecycle of the device. Trust Management and Classification (section 3.7.3) for allowing access to the trust broker. Policy Management (section 3.7.4) for managing all the policies around this process and on the device itself. Audit (section 3.7.6) to audit the controls and this process.
Name process: Device Monitoring	*Description:* The devices and their respective actions should be monitored all the time. This can either be done by the enterprise that owns the device or by a federated approach or later on, the trust broker. All the actions that lower the trustworthiness of the device should be taken into account while monitoring. The security status should be checked continuously as well by an agent on the device and the trust broker itself. Checks should be made for the right security software and secure behaviour of the device (e.g. no malware related actions should be executed by the device). This process should trigger the Device Management process if necessary.
	Interacts with following services: Identity Management, Federation and Reputation (section 3.7.2) will be used for the Identity Lifecycle of the device. Trust Management and Classification (section 3.7.3) for allowing access to the trust broker. Policy Management (section 3.7.4) for managing all the policies around this process and on the device itself. Audit (section 3.7.6) to audit the controls and this process.
Name process: Device Management	*Description:* The devices should be managed mostly by the agent and the user himself. Based on reports of the Device Monitoring process, actions should be taken such as software updates, replacement of hardware(parts) et cetera.
	Interacts with following services: Identity Management, Federation and Reputation (section 3.7.2) will be used for the Identity Lifecycle of the device. Trust Management and Classification (section 3.7.3) for allowing access to the trust broker. Policy Management (section 3.7.4) for managing all the policies around this process and on the device itself. Audit (section 3.7.6) to audit the controls and this process.
Name process: Device Off-boarding	*Description:* The device is no longer necessary for the domain. The local data will either be recovered and loaded into other media or destroyed. Keys may need to be repudiated and potentially any software that was loaded onto that device as part of the on-boarding and/or management process removed and de-licensed. This process will trigger the device de-provisioning if the device is from the same company as where the off-boarding happens. The extra information of the device (its identity et cetera) will be archived.

	Interacts with following services: Identity Management, Federation and Reputation (section 3.7.2) will be used for the Identity Lifecycle of the device. Trust Management and Classification (section 3.7.3) for allowing access to the trust broker. Policy Management (section 3.7.4) for managing all the policies around this process and on the device itself. Audit (section 3.7.6) to audit the controls and this process.
Name process: device De-provisioning	*Description:* if the device is no longer necessary then it will be de-provisioned. All additional software and data will be removed. What will happen after that is organisation specific. The Identity of the device will be recorded as de-provisioned.
	Interacts with following services: Identity Management, Federation and Reputation (section 3.7.2) will be used for the Identity Lifecycle of the device. Trust Management and Classification (section 3.7.3) for allowing access to the trust broker. Policy Management (section 3.7.4) for managing all the policies around this process and on the device itself. Audit (section 3.7.6) to audit the controls and this process.
Name process: Device Lock out	*Description:* If the device needs to be remediated by the device management process, then the device can be locked out form the transaction / collaboration process until it is remediated. It will be locked of all transactions (via (say) a change of rules to its personal firewall) until it has been remediated.
	Interacts with following services: Identity Management, Federation and Reputation (section 3.7.2) will be used for the Identity Lifecycle of the device. Trust Management and Classification (section 3.7.3) for allowing access to the trust broker. Policy Management (section 3.7.4) for managing all the policies around this process and on the device itself. Audit (section 3.7.6) to audit the controls and this process.
Name process: Software Provisioning	*Description:* This could be either the direct follow-up process of the Device Provisioning or Device On-boarding process. The software will be licensed and distributed to the device. It will be registered as provisioned to the device ID and the user ID in both the trust brokers' repository and the enterprise repository. It will further more trigger the Software Planning and Monitoring processes.
	Interacts with following services: Identity Management, Federation and Reputation (section 3.7.2) for managing the Identity Lifecycle and registration processes for both the user and the device. Trust Management and Classification (section 3.7.3) for linking to the trust broker repository and for classifying the information that comes with the process. Policy Management (section 3.7.4) for applying the new policies that come with the software. Meta/Information Management (section 3.7.5) for managing the information that comes with the software. Audit (section 3.7.6) for auditing the (controls of this) process.
Name process: Software Planning	*Description:* The rest of the software lifecycle (monitoring, maintenance and de-provisioning) will be planned in advance. The information of the manufacturer of the software around patches and updates will be used combined with the information of both the People Lifecycle Management and Device Lifecycle Management to plan and schedule the complete Software Maintenance and Software de-provisioning process.
	Interacts with following services: Identity Management, Federation and Reputation (section 3.7.2) for managing the Identity Lifecycle for both the user and the device. Trust Management and Classification (section 3.7.3) for linking to the trust broker repository and for classifying the information that comes with the process. Policy Management (section 3.7.4) for applying planning policies. Meta/Information Management (section 3.7.5) for

	managing the information that comes with the planning process. Audit (section 3.7.6) for auditing the (controls of this) process.
Name process: Software Monitoring	*Description:* This process monitors the behaviour of the software. It checks for failures, errors or behaviour in violation with policies. The findings of this process will be recorded and saved. They will also be used for Device Monitoring if necessary.
	Interacts with following services: Identity Management, Federation and Reputation (section 3.7.2) for managing the Identity Lifecycle for both the user and the device. Trust Management and Classification (section 3.7.3) for linking to the trust broker repository. Policy Management (section 3.7.4) for monitoring policies. Audit (section 3.7.6) for auditing the (controls of this) process.
Name process: Software Maintenance	*Description:* This process is being executed whenever an update/patch or other maintenance work will have to be done to the software. This can be highly automated and partly done by the user who owns the device. It is triggered by the Software and Device Monitoring processes as well as the Software Planning process.
	Interacts with following services: Identity Management, Federation and Reputation (section 3.7.2) for managing the Identity Lifecycle for both the user and the device. Trust Management and Classification (section 3.7.3) for linking to the trust broker repository and for classifying the information that comes with the process. Policy Management (section 3.7.4) for applying new policies through the maintenance process. Meta/Information Management (section 3.7.5) for managing the information that comes with the process. Audit (section 3.7.6) for auditing the (controls of this) process.
Name process: Software de-Provisioning	*Description:* If the software is in the End of Lifecycle stage or if the device is off-boarded or de-provisioned, then this process will be executed. Software will be removed, de-licensed and important information will be archived or destroyed.
	Interacts with following services: Identity Management, Federation and Reputation (section 3.7.2) for managing the Identity Lifecycle for both the user and the device. Trust Management and Classification (section 3.7.3) for linking to the trust broker repository and for classifying the information that comes with the process. Meta/Information Management (section 3.7.5) for managing the information that comes with the process. Policy Management (section 3.7.4) for applying new policies through the de-provisioning process. Audit (section 3.7.6) for auditing the (controls of this) process.

Table 12: Processes description of Device Lifecycle Management.

Requirements:

The following requirements can be defined for the Device Lifecycle Management processes, based on sections 2.6.4(protocols and standards) 2.6.5(Policy management), 2.6.6 (data classification, privacy and protection), 2.6.7 (Identity Management), 2.6.8 (trust, -management), 2.6.9 (End-point security), 2.6.10 (Audit), the Jericho Forum Commandments and (Forum 2008a):

- Requirements derived from protocols and standards:
  - The devices should be equipped at least with firewalls, the proper message protection mechanisms, web service filters et cetera.
  - There should not be any usage of the IPSEC and SSL tunnels between the end-point and the server. That is only allowed between servers in a static environment.

- Requirements derived from policy management[114]: see also Information Lifecycle Management. Additional requirements:
  - All end-points should be capable of maintaining the security policies throughout the complete set of processes and in all environments (i.e. inside the corporate environment and outside the corporate environment).
- Requirements derived from data classification[115]: see Information Lifecycle Management, the same requirements hold.
- Requirements derived from privacy: see Information Lifecycle Management, the same requirements hold.
- Requirements derived from data protection: see Information Lifecycle Management, the same requirements hold.
- Requirements derived from Identity Management[116]: see Information Lifecycle Management, the same requirements hold and the following additional requirements:
  - The identity system should also work for devices with the same Identity Lifecycle.
  - The processes should be compatible with multiple types of identity systems.
- Requirements derived from trust and -management[117]: see People Lifecycle Management, the same requirements hold and the following additional requirements:
  - All devices should be capable of connecting to the trust management framework and the trust broker.
  - All devices should be registered at a trust broker.
- Requirements derived from Endpoint security:
  - The Endpoint Security should be implemented by these processes in such a way that it is capable of communicating his trust status at all times to different collaborative entities (with or without the trust broker).
  - The Endpoint Security should be implemented by these processes in such a way that it is capable of fully protecting the device that it is operating on.
  - The processes should allow scalable and manageable End-point security solutions.
  - The End-point Security should be implemented by these processes in such a way that mutual trust is supported.
  - The End-point Security should be implemented by these processes in such a way that that there is no single point of failure.
  - The End-point Security should be implemented by these processes in such a way that it can work with all open and inherently secure protocols on the field of security.
  - The processes of Device Lifecycle Management should implement segregation of duties.
  - The End-point Security should be implemented by these processes in such a way that it allows one to apply rights to external devices accessing end-points under his control.
  - The processes of Device Lifecycle Management should interact with Identity Lifecycle Management in order to supply the necessary information for the Information Lifecycle Management processes.
  - The End-point Security should be implemented by these processes in such a way that it will allow multiple types of architectural implementations until the trust broker can be used as the supporting architecture.
  - The End-point Security should be implemented by these processes in such a way that it can deliver all the necessary data for data classification, available encryption methodology and accountability processes.

[114] See for the other requirements derived from / on Policy management section 3.7.4.

[115] See for the other requirements derived from / on Data classifcation section 3.7.3.

[116] See for the other requirements of Identity Management section 3.7.2.

[117] See section 3.7.3 for the other requirements surrounding Trust and -management.

  - The End-point Security should be implemented by these processes in such a way that all devices should have a high availability, flexibility and still be secured[118].
- Requirements derived from Audit[119]: see People Lifecycle Management, the same requirements hold.
- Requirements derived from the Jericho Forum Commandments**:**
  - The scope of the processes should be exactly on all the devices that are necessary for the collaborative relationship (and no more or less) (JFC1).
  - The protective measures that will be taken during the processes must always be appropriate to the device, the transactions and the information assets on the device (JFC1).
  - The processes should be easy to manage, scalable and simple (JFC2).
  - The processes and devices should continue to function, even on the internet (JFC3 and JFC5).
  - The processes should work based on open and secure protocols (JFC4).
  - The processes should manage the devices in such a way that they all will have a determinable trust level at each given time (JFC6&7).
  - The Device Lifecycle Management processes should also be executed outside the locus of your control (JFC8).
- Requirements derived from (Forum 2008a):
  - The management of devices must function identically irrespective of whether a device is connected to the Intranet or Internet.
  - All protocols involved in the management of the device must be inherently secure.
  - Software management should cover all software (OS, BIOS and application software).
  - The On-boarding process should be capable of being executed ad-hoc so that alien devices can quickly be added to the collaborative relationship if necessary.
  - The device needs to be capable of positively and uniquely identifying itself to other systems in a form that cannot be subverted.
  - The Device on-boarding and Device monitoring processes should check for each device the end state and capability, combined with the (automated) risk assessment for an assessment of a device's suitability.
  - The device Off-boarding process should use a re-validation step on the device itself, which will need human interaction to ensure proper off-boarding.
  - The processes should be executed by either the owner of the device or a federated authority.
  - The end-point security check should not be done by a tunnelled connection or at a network gateway.
  - The End-point security standard that needs to be developed needs to be vendor neutral.

Recommendations:

The following recommendations for Device Lifecycle Management can be made based on chapter 2, paragraph 3.3 and this section:

- **Compatibility with SOI:** in order to make the processes compatible with the infrastructures of today, one should make sure that it can be easily adopted by companies that use SOI by making it compatible to SOI..
- **Service based implementation:** the process should be implemented based on services, allowing one to take the full benefits of the available SOA.
- **Development alongside service development:** the processes should be developed together with its most important services such as the Identity Management, trust management and

[118] See for more details section 2.6.9.

[119] See for the other requirements derived from / on Audit section 3.7.6.

end-point security services. This will require parallel roadmaps to the services to ease such development.

- **Complete agent solution:** vendors should try to create a security agent, which is capable of monitoring every named aspect of the device, and communicate the status to the user and the trust broker. The agent should be non-falsifiable to allow multiple types of end-point security architectures as seen in section 2.6.9.
- **Additional lifecycle management process:** The Software Lifecycle Management should be separated from the Device Lifecycle Management processes in order to allow a better research base and a more specific approach of the lifecycle.
- **Additional governance:** an additional governance framework should be designed for auditing the processes and providing the right governance tools for the management of them.
- **The creation of an open standard:** the processes described here should be detailed and standardised in an open and inherently secure standard that is protocol and model independent and allows multiple vendors to come up with solutions that will be interoperable.
- **Further research of the processes:** all of these processes should be further researched in order to be capable of developing solutions for them. They should be detailed and prepared for standardisation.

COA V2.0 UPDATES:

The following relevant additions have been made in the COA 2.0 revision:

- **Device Lifecycle Management**, in: "COA Processes Device Lifecycle Management", which can be summarized as follows:
  - *Software management must cover all software.* All of the software on the devices should be managed by this process.
  - *No assumptions about registration origin*. There should be no assumption in the registration phase of where the device is registering from.
  - *Additional factors that dictate device interaction*. A set of additional factors have been released, such as: amount of memory, presence of a particular application, how the connection is being made, the network speed/ or cost of connection.
  - *Lockout until remediation.* A device must stay locked out until it is remediated.

The publications can be found at: https://www.opengroup.org/jericho/publications.htm

3.6.6. Enterprise Relationship Management

Introduction:
Enterprise Relationship Management consists of processes that are focussed on initiating, operating, and closing down collaborations. It also includes mapping the critical relationships between all the collaborating parties and identifying the most crucial, valuable and endangering relationships. The risk and the value of a relationship are the most important themes in all of these processes. (Forum 2008e)

The description, requirements and recommendations of these processes will be based on (Forum 2008b; Forum 2008e), the COA Principles, SLATES and the knowledge found in chapter 2.

Intermezzo 10: The rationale behind Enterprise Relationship Management

"The rate of change of external enterprise relationships is accelerating and out pacing the traditional means of management. External business relationships are more often created across the internet, which enables a level of agility that is hard to match with manual relationship management processes. Business drivers are requiring organisational transformations to occur in hours rather than months. Individuals within organisations have the power to create external relationships with just one click. The processes to manage the life cycle of such relationships, at these speeds, are immature at best and often non-existent."..."The increasing number of external relationships and the high reliance that enterprises are putting on such relationships, coupled with the immaturity of the management processes has the potential to grow an enterprises relationship risks to unacceptable levels. Litigation in the future will likely be founded upon the lack of management oversight of relationships that management were not even aware existed."

(Forum 2008b)

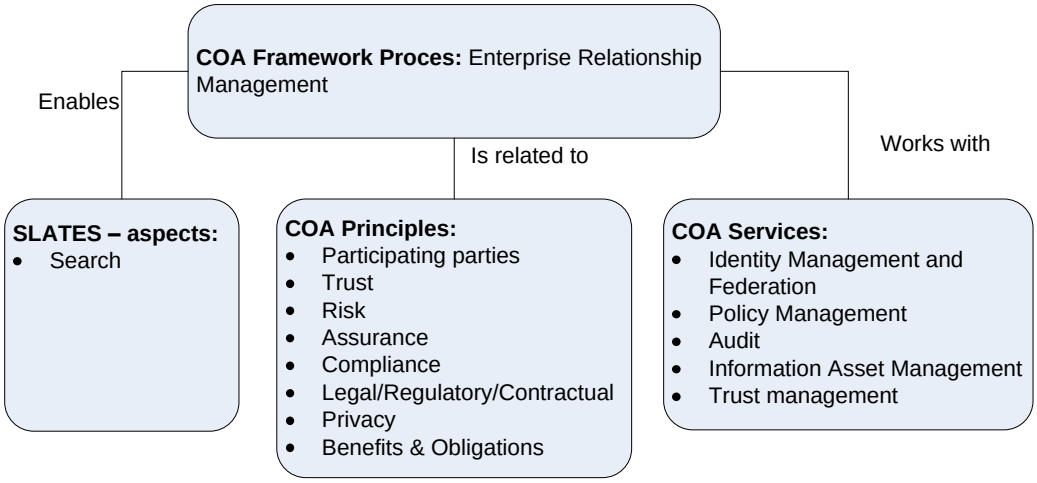


Figure 58: Relations between Enterprise Relationship Management, SLATES, Principles and Services.

Relation to SLATES

Enterprise Relationship Management allows organisations to protect their relationships and safely disclose information onto one another (with the help of the Information Lifecycle Management processes). By disclosing the information domains to one another, one can easily index the domains and make the information searchable.

All of the other SLATES aspects are supported indirectly by the Enterprise Relationship Management Processes.

Relation to the COA Principles

There are multiple relations with the COA Principles, this process set connects to or supports the following principles:

- **Participating parties:** in order to understand with whom one is transacting, the relationship itself will have to be identified as well. Enterprise Relationship Management will allow a better management of the identities of those outside one's own organisation.
- **Trust:** in order to understand the necessary level of trust that is required for a transaction, one will have to understand the relationship between the transacting entities. Yet, in order to truly understand the necessary level, one will have to oversee all of the stakeholders and different roles and relations inside the collaborative network where the transaction is taking place. See section 2.5.4 for more details.

- **Risk:** assessing the risk of the environment, also means assessing the risk coming from relations and relations between those with one has a relationship (see section 2.5.4 for more details).
- **Assurance:** the artefacts that are produced during the execution of these processes will often provide assurance for the required trust level.
- **Compliance**: the processes that manage the relationships will have to use the compliancy data from others to see whether the relations can be endangering.
- **Legal/Regulatory/Contractual:** the relationships can be protected by legal, contractual and regulatory means. At the other hand: some of the legal means can have either a positive or a negative impact on the benefits of a certain relationship.
- **Privacy:** as loads of PII will be processed by these processes, the principle of Privacy should be taken in mind as well.
- **Benefits and Obligations:** many of the aspects of this principle will influence the relationships and the need for having them such as good corporate citizenship and customer relationships.

Relation to the services defined in the COA framework

The Enterprise Relationship Management processes can be related to the following COA framework Services:

- **Identity Management and Federation**: in order to identify the persona of the other organisations and have the capability of checking their reputation, one will have to use the Identity Management services.
- **Policy Management:** this service can be used to manage cross-organisational policies in order to protect the relationship.
- **Audit:** one will have to audit one another to see whether everything is still in such a shape that the relationship will provide benefits instead of risks and danger. Furthermore, one will have to audit each process (control) to see whether it is still performing well.
- **Trust management and Classification:** for classifying all the assets that will be exchanged between the parties in the collaborating environment and/or produced and/or used by the processes. Furthermore, it will be used for connecting to the trust broker.
- **Information Asset Management:** for managing all the assets that will be exchanged between the parties in the collaborating environment and/or produced and/or used by the processes.

A short description of the process:

The processes as defined in (Forum 2008b) have been used as the primary directive for defining the processes. Other sources that have been used: paragraphs 2.5 and 3.3:

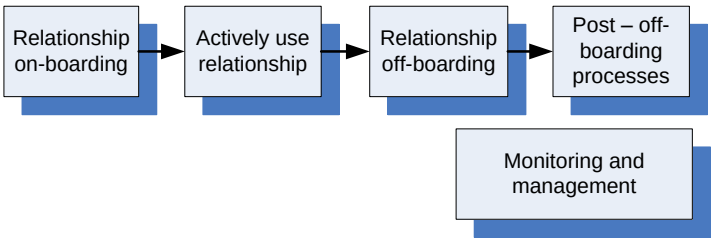


Figure 59: Enterprise relationship management (simplified).

Name: Enterprise Relationship Management (ERM)	
Description: ERM is aimed at identifying new relationships and manage existing relationships. This is done by several processes: one will need environmental monitoring to see if there are more interesting parties out there, Furthermore, the current collaborative network needs to be monitored and then the relationship lifecycle needs to be managed.	
Name Process: Environmental monitoring	*Description:* this process checks the environment outside the collaborative network for interesting new upcoming parties and products that could help the current collaborative environment in realising its goals. It will also check for new potential endangering parties and products and register all of the information.
	Connects with the following services: Meta/Information Management (Section 3.7.5) for managing the information that will be created by this

	process. Audit (section 3.7.6) for auditing this process and its controls. Trust Management and Classification (section 3.7.3) for classifying the information that will be created by this process.
Name Process: Collaborative Network Monitoring	*Description:* This process checks the collaborative network for new parties, checks the relations between the current parties and assesses them as valuable, endangering et cetera. All direct and indirect relationships will be assessed. The data will be used for the relationship management or relationship on- and off-boarding processes. It will also check which role each party has in the network (see also section 2.5.4).
	Connects with the following services: Meta/Information Management (Section 3.7.5) for managing the information that will be created by this process. Audit (section 3.7.6) for auditing this process and its controls. Trust Management and Classification (section 3.7.3) for accessing the trust broker for retrieval of information about the current collaborative network and its entities and for classifying the information that will be created by this process.
Name Process: Relationship On-boarding	*Description:* this process starts an active collaborative relationship by the following steps: 1.Collaborating Party Identified: Creates a new entry in the collaborating party directory. 2.Relationships Identified: The likely relationships that will be involved in the collaboration are identified, classified and documented. 3.Regulations and Policies Identified: The likely regulations and policies affecting the relationships and its parties are identified and documented. 4.Outgoing Information Assets Identified: The information assets that will be transferred to the collaborating party are identified, classified and documented. 5.Incoming Information Assets Identified: The information assets that will be transferred from the collaborating party are identified, classified and documented. 6.Enterprise Risks Identified: The key enterprise risks that may be affected by the various relationships with the collaborating party. 7.Business Impacts Identified: The potential likely Business Impacts that may occur to either or both parties as a result of the relationships, resulting in the documentation of the impacts and the classification of the relationships 8.Initial Personnel On-Boarded: see process person on-boarding from People Lifecycle Management. 9.Physical and System Access: The access requirements are identified and provisioned associated with each person based on the relationships they are engaged in. 10.Contractual Obligations and Control Objectives: Creates and electronically documents the Contractual Obligations and required Control Objectives between the parties including terms of cessation. (Forum 2008b)
	Connects with the following services: Meta/Information Management (Section 3.7.5) for managing the information that will be created by this process. Audit (section 3.7.6) for auditing this process and its controls. Policy Management (section 3.7.4) for managing the policies that are involved in this process. Identity Management, Federation and Reputation (section 3.7.2) for managing the identities of new personnel. Trust Management and Classification (section 3.7.3) for managing the link with the trust broker in saving all the new information and for classifying the information that will be created by this process.
Name Process:	*Description:* this process monitors only the active relationships. It checks for

Relationship monitoring	current shared employees, resources, policies, the operational risk and if the behaviour of the related parties is in line with contractual, legal and regulatory means.
	Connects with the following services: Meta/Information Management (Section 3.7.5) for managing the information that will be created by this process. Audit (section 3.7.6) for auditing this process and its controls. Policy Management (section 3.7.4) for managing the policies that are involved in this process. Identity Management, Federation and Reputation (section 3.7.2) for managing the identities of the personnel of the other companies. Trust Management and Classification (section 3.7.3) for managing the link with the trust broker in saving all the new information for classifying the information that will be created by this process.
Name Process: Relationship management	*Description:* based on the findings of the previous monitoring processes, this process will allow the personnel of the company to get into action. The actions can vary from off- or on-boarding personnel based on changes in the relation, to taking legal actions due to not following the contractual obligations.
	Connects with the following services: Audit (section 3.7.6) for auditing this process and its controls. Policy Management (section 3.7.4) for managing the policies that are involved in this process. Identity Management, Federation and Reputation (section 3.7.2) for managing the identities of the personnel of the other companies if the actions require doing so. Trust Management and Classification (section 3.7.3) for managing the link with the trust broker in saving all the new information.
Name Process: Asset monitoring	*Description:* The information assets that are used in the collaborative relationship will be monitored. A record of all of the exchanged/transferred information assets will be maintained by this process.
	Connects with the following services: Meta/Information Management (Section 3.7.5) for managing the information that will be created by this process. Audit (section 3.7.6) for auditing this process and its controls. Policy Management (section 3.7.4) for managing the policies that are involved in this process and for classifying the information that will be created by this process.
Name Process: Relationship Off-boarding	*Description:* the off-boarding consists of initiating the Person Rollout, the Device Off-boarding and Information destruction process (for destroying those assets that one is not allowed to use after the collaborative relationship due to contractual means). Which processes exactly will be triggered and which additional processes will be started are depend on the fact whether the enterprise still has active relations to the collaborative network.
	Connects with the following services: Meta/Information Management (Section 3.7.5) for managing the information that will be handled by this process. Audit (section 3.7.6) for auditing this process and its controls. Policy Management (section 3.7.4) for managing the policies that are involved in this process. Identity Management, Federation and Reputation (section 3.7.2) for managing the identities of the personnel that might be de-provisioned. Trust Management and Classification (section 3.7.3) for managing the link with the trust broker in saving all the new information.
Name Process: Off-boarding Review	*Description:* after the off-boarding an Off-boarding Review will be necessary. This process will check if all of the assets, identities, relations, devices et cetera are handled well according to contractual, legal and regulatory means. The process will check the all the steps of the on-boarding process in reverse.
	Connects with the following services: Meta/Information Management (Section 3.7.5) for managing the information. Audit (section 3.7.6) for

	auditing this process and its controls and for auditing the result of the previous processes. Policy Management (section 3.7.4) for managing the policies that are involved in this process. Identity Management, Federation and Reputation (section 3.7.2) for reviewing the output of the service considering the other identities. Trust Management and Classification (section 3.7.3) for managing the link with the trust broker in saving all the new information and reviewing the current information and for classifying the information that will be created by this process.
Name Process: Off-boarding Post processing	*Description:* this process will be triggered by the Off-boarding Review in order to handle the issues that have been found in the reviewing process.
	Connects with the following services: Meta/Information Management (Section 3.7.5) for managing the necessary information. Audit (section 3.7.6) for auditing this process and its controls. Policy Management (section 3.7.4) for managing the policies that are involved in this process. Identity Management, Federation and Reputation (section 3.7.2) for managing the identities of the personnel when needed. Trust Management and Classification (section 3.7.3) for managing the link with the trust broker in saving all the new information and for classifying the information that will be created by this process.

Table 13: Processes description of Enterprise Relationship Management.

Requirements:

The following requirements can be defined for the Enterprise Relationship Management processes, based on paragraph 2.5, sections 2.6.5 (Policy Management), 2.6.6 (Privacy, Information classification and information protection), 2.6.7 (Identity Management), 2.6.8 (Trust and –management), 2.6.10 (Audit), the Jericho Forum Commandments and (Forum 2008b):

- Requirements derived from collaboration:
  - The entire network should be mapped.
  - The processes should be capable of mapping multiple roles inside different networks to a certain party.
  - The processes should be capable of maintaining and monitoring the position of the organisation itself in multiple networks.
  - The processes should be usable in any kind of internet based collaboration form.
- Requirements derived from Policy management[120]: see also Information Lifecycle Management.
- Requirements derived from Privacy: see also Information Lifecycle Management. Additional requirements:
  - PII should only be stored about other organisational members if strictly necessary.
- Requirements derived from Data classification[121]: see also Information Lifecycle Management.
- Requirements derived from Data protection[122]: see also Information Lifecycle Management.
- Requirements derived from the Identity Management and related fields[123]: see also Device Lifecycle Management. Additional:
  - The data of the Identity Management should be handled with care according to legal and regulatory law.

[120] See for the other requirements derived from / on Policy management section 3.7.4.

[121] See for the other requirements derived from / on Data classifcation section 3.7.3.

[122] See for the other requirements derived from / on Data protection section 3.7.5.

[123] See for the other requirements derived from / on Identity Management section 3.7.2.

- Requirements derived from Trust, -management and –brokers[124]: see also People Lifecycle Management.
- Requirements derived from Audit[125]: see also People Lifecycle Management. Additional:
- Requirements derived from the Jericho Forum Commandments:
  - The measures used by the processes to protect or to optimise the relationship should be appropriate to the asset at risk (JFC1).
  - The processes should be scalable over multiple networks and easy to manage (JFC2).
  - The processes should perform under any circumstances in order to allow collaboration in multiple environments and states of the enterprises and their relationships (JFC3&4).
- Requirements derived from (Forum 2008b):
  - All collaborating parties should be identified.
  - Direct relationships should be identified and categorised Critical or Non Critical.
  - Indirect relationships should be identified where they are likely to be critical to the operation of the collaboration.
  - Critical relationships should be mapped across the various entities involved.
  - Critical relationships contract obligations should be documented, preferably using a standard electronic format.
  - The business risks related to these relationships should be identified and documented.
  - The information assets involved in critical relationships should be identified, classified, and documented.
  - The other PRIDE principles, practises and processes should be used in conjunction with these principles, practises and processes.

Additional: COBITs DS2 for Enterprise Relationship Management

As described in section 2.4.7 and in chapter 1, there would be a little experimenting with DS2: Manage Third-party Services. It has been described as:

> *"The need to assure that services provided by third parties (suppliers, vendors and partners) meet business requirements requires an effective third-party management process. This process is accomplished by clearly defining the roles, responsibilities and expectations in third-party agreements as well as reviewing and monitoring such agreements for effectiveness and compliance. Effective management of third-party services minimises the business risk associated with non-performing suppliers."*
>
> (Institute 2007a)

It is focussed on the IT process that should manage the third-party service. It tries to satisfy the business requirement for IT of "providing satisfactory third-party services while being transparent about benefits, costs and risks". It is achieved by identifying and categorising supplier services, identifying and mitigating supplier risk, monitoring and measuring supplier performance. It is measured by number of user complaints due to contracted services, percent of major suppliers meeting clearly defined requirements and service levels, percent of major suppliers subject to monitoring. (Institute 2007a)

It Furthermore, gives the following four control objectives:

> *"DS2.1 Identification of All Supplier Relationships*
> *Identify all supplier services, and categorise them according to supplier type, significance and criticality. Maintain formal documentation of technical and organisational relationships covering the roles and responsibilities, goals, expected deliverables, and credentials of representatives of these suppliers.*
> *DS2.2 Supplier Relationship Management*

[124] See section 3.7.3 for the other requirements surrounding Trust and -management.

[125] See for the other requirements derived from / on Audit section 3.7.6.

Formalise the supplier relationship management process for each supplier. The relationship owners should liaise on customer and supplier issues and ensure the quality of the relationship based on trust and transparency (e.g., through SLAs).
DS2.3 Supplier Risk Management
Identify and mitigate risks relating to suppliers' ability to continue effective service delivery in a secure and efficient manner on a continual basis. Ensure that contracts conform to universal business standards in accordance with legal and regulatory requirements. Risk management should further consider non-disclosure agreements (NDAs), escrow contracts, continued supplier viability conformance with security requirements, alternative suppliers, penalties and rewards, etc.
DS2.4 Supplier Performance Monitoring
Establish a process to monitor service delivery to ensure that the supplier is meeting current business requirements and continuing to adhere to the contract agreements and SLAs, and that performance is competitive with alternative suppliers and market conditions."

(Institute 2007a)

As a last addition, before we will discuss the topic, let us look at the optimised state of the process. It has been described as:

"Optimised when Contracts signed with third parties are reviewed periodically at predefined intervals. The responsibility for managing suppliers and the quality of the services provided is assigned. Evidence of contract compliance to operational, legal and control provisions is monitored, and corrective action is enforced. The third party is subject to independent periodic review, and feedback on performance is provided and used to improve service delivery. Measurements vary in response to changing business conditions. Measures support early detection of potential problems with third-party services. Comprehensive, defined reporting of service level achievement is linked to the third-party compensation. Management adjusts the process of third-party service acquisition and monitoring based on the measurers."

(Institute 2007a)

A good question would be "Does DS2 cover the needs for Enterprise Relationship Management?".
The answer is "no", based on the following:
The Enterprise Relationship Management will manage all of the relationships inside a collaborative network. It will also check and map all the relationships inside the network, which are not connected to the organisation that executes the processes for itself inside its COA.
The DS2 is focussed on "suppliers". Of course, if a supplier is seen as "a value adding party" then one might say that it will cover all the parties for the direct relations. Yet the indirect relations, the mapping and identification of the relationships themselves are of scope of DS2.
That is why DS2 will not cover the Enterprise Relationship Management and thus additional Governance measures will have to be designed in order to apply proper governance over these processes.

Recommendations:
The following recommendations for Enterprise Relationship Management can be made based on chapter 2, paragraph 3.3 and this section:

- **Additional governance:** an additional governance framework should be designed or chosen and re-designed for auditing the processes and providing the right governance tools for the management of them. COBIT DS2 could be used as a starting point for the Enterprise Relationship Management processes and then should be extended until it covers all of the Enterprise Relationship Management processes.
- **The creation of an open standard:** the processes described here should be detailed and standardised in an open and inherently secure standard, which is protocol and model

independent and allows multiple vendors to come up with solutions that will be interoperable.

- **Further research of the processes:** all of these processes should be further researched in order to be capable of developing solutions for them. They should be detailed and prepared for standardisation.

COA V2.0 UPDATES:

The following relevant additions have been made in the COA 2.0 revision:

- **Structural changes**, in: "Position Paper – COA Framework", which can be summarized as follows:
  - *Enterprise relationship management refered to as called Enterprise Management:* in both this and the paper "Enterprise Lifecycle Management" the Enterprise Relationship Management Enterprise is also refered to as Enterprise Lifecycle Management.
- **Additional recommendations**, in "Enterprise Lifecycle Management", which can be summarized as follows:
  - *Recommendation to raise effectiveness and efficiency of the procedures*: they recommend to shorten the time of setting up the time needed to setup and close down a collaboration, since lengt of time of a collaboration itself is rapidly falling.

The publications can be found at: https://www.opengroup.org/jericho/publications.htm

3.6.7. Summary: the COA Processes

Looking back at this paragraph, we can summarise our findings of the COA Processes as follows:

- **People Lifecycle Management** is about managing the complete lifecycle of the employees inside the collaborative network. It consists of on- boarding, monitoring, maintenance and off- boarding. It uses all of the COA framework Services, is an important base for many COA Principles and helps enabling the Authorship of SLATES. However, the processes are far from finished. They are still in design stage. We have set up a set of requirements based on background research and the first Jericho Paper. Further research for detailing the processes, standardising them and the release of a Process Paper are highly recommended.
- **Risk Management** is about assessing the information risks involved in collaborations. It is a very important process, since it will support all COA Principles. It will also work with a set of services as well. However, more research should be done to detail these processes and standardise them.
- **Information Lifecycle Management** is about managing the complete lifecycle of information from creation, spreading, access, reading, updating, destruction and anything else that could happen to the information. This process will maximise the availability of SLATES, since it will support all of the aspects. It works with all the services defined in the COA framework and supports many principles. However, the processes are far from finished. They are still in design stage. We have set up a set of requirements based on background research and the first Jericho Paper. Further research for detailing the processes, standardising them and the release of a Process Paper are highly recommended.
- **Device Lifecycle Management** is about managing both the lifecycle of the devices and the software on it. Therefore, there are actually two lifecycle management processes there. The first is focussed on maintaining the trust state by managing the cycle that consists of provisioning, on-boarding, monitoring, management, off-boarding, de-provisioning and lock out. The second is focussed on the software lifecycle, which consists of software provisioning, planning, monitoring, maintenance and de-provisioning. Both these lifecycles do support SLATES indirectly, use all the services defined in the COA framework and support most of the COA Principles. However, the processes are far from finished. They are still in design stage. We have set up a set of requirements based on background research and the first Jericho

Paper and the process paper. Further research for detailing the processes, splitting the software lifecycle management and the Device Lifecycle Management processes and standardising them are highly recommended.

- **Enterprise Relationship Management** is focussed on managing the relationships inside the collaborative network. It consists of environmental monitoring, collaborative network monitoring, relationship on-boarding, monitoring, management ,off-boarding, asset monitoring and an additional off-boarding review with post off-boarding processes. These management processes could support Search from SLATES by opening up each other's enterprise information domain. All the COA Principles are supported by these processes and it relies on a few COA Services as well. However, the processes are far from finished. They are still in design stage. We have set up a set of requirements based on background research, the first Jericho Paper and the process paper. Further research for detailing the processes and standardising them are highly recommended.

3.7.COA services

3.7.1.Introduction

The next step in the detailing process is detailing the COA framework Services. As we have already detailed the COA Principles and Processes, we have seen that many of them rely on these services.
Each service will be detailed a little based on this and the previous chapter. A short description will be given, a set of requirements and a set of recommendations. However, if one wants to know more about a specific service, then he should check chapter two and the quoted literature.
As one can understand, there will be plenty of relationships between the different services. However, they are of scope of this thesis for now.

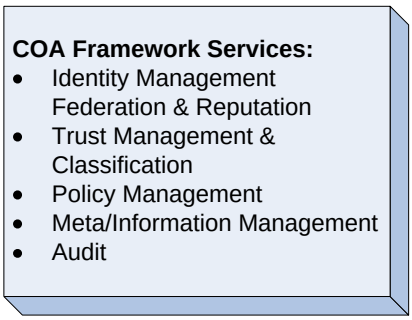


Figure 60: The COA framework Services.

3.7.2.Identity Management, Federation and Reputation

Introduction
The Identity Management, Federation and Reputation service will be very important, it is one of the basics where all the processes from People Lifecycle Management and Device Lifecycle Management rest upon. It is also of great value to the other processes (Risk-, Information lifecycle- and Enterprise Relationship Management).
The description, requirements and recommendations in this section will be derived from sections 2.6.7 (Identity Management, user authentication and federation), 2.6.9(Privacy), section 2.6.6 (End-point Security) and paragraphs 3.3(COA framework overview) 3.5 (COA Principles), 3.6 (COA Processes). See also the named paragraphs and sections for more details.

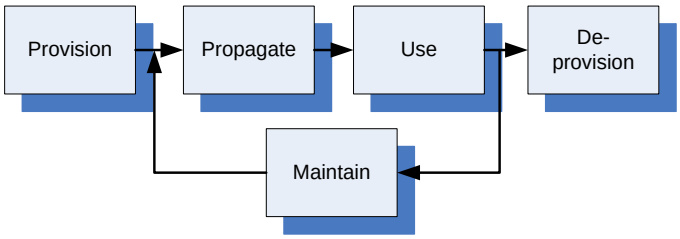


Figure 61: Digital Identity Lifecycle (based on (Barannikov 2008)).

Description:
The Identity Management service will take care of the complete Identity Lifecycle of all devices and people in the collaborative environment. As any other SOA based entity, it

will be based on a set of services that provide the total service as described here. See section 2.2.2 for more details.
The lifecycle starts with provisioning: a device or a person will be provisioned with an identity. The lifecycle management processes will trigger the service to start the provisioning by starting the verification processes. The verification and provisioning process will differ depending on the type of device, the type of person and the (collaborative environment or) organisation that establishes it. If an organisation does not feel like using low risk/high trust values, then it does not have to take every measure there is for identity verification. After that, the identity will propagate in such a way as the type of identity supports[126]. From there on it can be used by the user or the device in authentication (often two way), authorisation and accounting processes. In most of those authorisation processes, the identity of the device and the identity of the person will be very important. A certain person will only be allowed to use a certain information asset based on his identity/role/reputation and the device he is using. The latter can be unravelled as the status of the end-point security of the device; its capabilities et cetera. One important thing to understand is that the trust state and/or reputation of an internal employee/device will be dealt with by the services. The trust state/reputation of the external member/device will be managed by the trust management service.
Of course, the identity will have to be maintained. The identity attributes of both the user and the devices will change over time and will have to be altered. These attributes can by anything: for a user it can vary from capabilities, to the trust broker where he is registered for maintaining the reputation, from PII that changes to roles and other additional attributes. For devices, it can vary from capabilities, to end-point security status et cetera.
When the device is no longer needed or when the users will leave the collaborative relationship[127] then the identity will be de-provisioned.
As the laws of identity show, the user should be in control of the data. That is why most of the services that this service will provide, will stay in contact with the user either by the PII broker or the trust broker or directly, to allow the user give consent for new actions.
The service itself will first cover central identity and then evolve to user centric identity (using federated identity as a step in between). Its in- and output will be almost the same at all times, in order to maintain its usability for the other processes that use the service.
Getting back to authorisation. Even though authorisation can be seen as a part of Identity Management, the actual permission to the authorisation will not be given by the service. Who is going to authorise what, will depend on the permission and the asset where the permission is about. Permissions are not seen as part of an identity, they can only be derived from it.
The authentication process will be processed by the service. Multifactor authentication will be used if necessary, whereas network device authentication will happen in various ways and not just the MAC based authentication, since that is not unique and easy to forge. The service itself will allow multiple types of authentication models.
Last but not least, the service will log all of the actions that have been undertaken by the services for accounting.

Requirements:
The following requirements can be derived from the findings of chapter 2 and 3:

- The services should adhere to one language that can express the credentials of principals and associated attributes required for identification, authentication and authorisation decisions.
- The services should support identity federation and, as soon as the technology is available, user centric identity.
- The services should support the currently available identity and transport protocols.

[126] This can differ depending on the type of identity that is being used, a Federated identity will have another propagation scheme than an User centric identity.

[127] This depends on the type of identity that is used. User centric identity will not be de-provisioned as a Federated identity.

- The services should be reliable, auditable and easily manageable.
- The services should be cost-effective and flexible.
- The services should be easy to use.
- The services should fit in the collaborative environment and deliver a suitable solution.
- The services should allow additional information to be stored either by the Identity Management service or in their own meta-information repositories.
- The services should take the seven laws of identity into account.
- The data of the Identity Management should be handled with care according to legal and regulatory law.
- The services should be location and protocol insensitive: one should be capable to execute them from any location and with any set of protocols, which covers all of the other requirements.
- The services should rely on the trust management architecture and the accountability information from the audit services for the reputational data.
- The requirements of authorisation and authentication as one can see in section 2.6.7 should be taken in mind as well as they are part of the Identity Management and the people lifecycle processes.
- The services should handle the PII properly in coordination with information asset management and the trust management framework via Trust Management and Classification.
- The services should allow the users to hold the data attributes themselves.
- The identity system should also work for devices with the same Identity Lifecycle as for personnel.
- Device authentication should not be easily spoofed.
- The services should provide all of the Identity Management means that are necessary for Person- and Device Lifecycle Management processes.

One should notice that the list of requirements is not exactly complete. This is because of the complexity of the service. More research will be necessary on this field.

Recommendations:

The following recommendations can be made, derived from the findings of chapter 2 and 3:

- **Use current existing technology:** to ensure that the Identity Management service will be workable in the current situation, one should try to use currently existing technologies to implement the services. This should make it a workable system.
- **Implement the solution in an open flexible Service Oriented fashion:** to ensure that one can use several protocols/standards/ types of identities, one should implement the services loosely coupled with clear in- and output definitions per service in order to ensure interoperability between the services and thus the Identity Management (sub)systems.
- **Separate services for devices- and personnel identification:** some of the mechanisms for device Identity Management and authentication will differ from those for personnel. That is why one should implement those separately.
- **Roadmap - Prepare for user centric identity:** there should be a roadmap for the Identity Management services that is focussed on implementing a globally accepted, fully accountable, user centric Identity Management system. One should ensure that there is enough flexibility in the current solution to implement the user centric identity solution.
- **Ensure that privacy concerns will be met:** privacy concerns should be met in design, build and testing of the service. This in order to guarantee both employees and customers that their privacy will be maintained. One should use other services such as Trust Management and Classification for classifying the information and later on use a PII broker, Policy Management to ensure the right information access policies, Meta/Information Management to protect the information and Audit to check the controls and events to ensure that the PII will be handled correctly.

- **No biometrics:** the identity system should not use biometrics as one of the default authentication mechanisms. This will fail as it can be easily compromised.
- **Further research:** the service and the available/required technologies should be further researched in order to make it a workable service that can be implemented conform the requirements and recommendations.
- **Cross-COA-service research:** one should research what data should exactly be exchanged between the services inside the Identity Management service and between all of the COA framework services.
- **Create a set of inherently secure open standards:** to ensure that all of the implementations along the roadmap will be universally exchangeable, one should create a set of inherently secure and open standards, which will allow one to follow all of the requirements and recommendations written in this document.

3.7.3. Trust Management and Classification

Introduction
This service or actually set of services could be seen as the second cornerstone. The trust management could be seen as a separate service that will allow one to connect to the trust management framework and manage the trust levels of devices, identities and enterprises. The Information Classification service is focussed on (re)classifying the information assets when they are created, incoming, updated or combined.
Both of these services will be discussed together in this section and derived from sections 2.6.8 (Trust, Trustmanagement and Trust brokers), 2.6.9 (End-point security), 2.6.7 (Accountability) 2.6.6 (Data classification) and paragraphs 3.3 (COA framework overview) 3.5 (COA Principles), 3.6 (COA Processes). See also the named paragraphs and sections for more details.

Description – Trust management:
As we have seen in section 2.6.8, one will have to trust the trustee in the fact that he will have the right competencies, capabilities and good intentions. One will also have to have a reason to trust the trustee. This can be based on information of third parties or reputation (the trust broker) his own experiences with the trustee and what a group of entities is saying about the trustee – again reputation - (again the trust broker).
However, it is very important to check the credibility of the third party (trust broker) or third parties in order to make sure that those reputational statements are right.
Besides checking the credibility, one will also have to ensure that there is an ontological structure that allows one to check for the capabilities and other reputational statements in an easier way. The ontological structure should be universal and defined as a standard in order to allow one to understand it.
Other important parts of trust management are based on Risk Management (the Risk Management process). In order to be trustworthy, one will have to negate the risks for the other parties and in order to trust someone, one will have to take the proper measures in order to be capable of trusting someone knowing his reputation, capabilities et cetera.
There are various recommendations for a certain trust model or trust architecture. In all types, it is most important to ensure that one can check the reputation, create a contract that takes all the extra legal issues in account (which will be taken care of by the trust broker or created on a global scale) and the behaviours and obligation will be monitored. Both the behaviours and the obligations will have to be monitored continuously and re-evaluated to see whether the reputation and the obligations still hold.

Having briefly addressed some of the most important aspects of trust management, one should now take a better look at the trust management service itself. What will it take care of? It will

have a set of services that will behave as one service that will allow one to have the following capabilities:
First, it has to manage the internal trust, which means that it has to check the reputation of the internal devices and employees. This will be done by end-point security services and by the Identity Management and Federation COA framework Service.
The end-point security services will work by using the trust brokers to communicate their security status and to let their status be assessed by the trust broker. See for more details section 2.6.9 as the end-point security is not a part of the services that will be discussed here.
It will also have to take care of the internal accountability records and check if everything is still in order. Furthermore, it will check the outcomes of the Risk Management processes such as the results of the risk assessment and the Risk Management process. This in order to check if there are any issues that are not properly handled which could negatively affect the reputation of the devices and the personnel. Of course, it will also have to check if the reputation has improved based on risk managing actions that have improved the reputation of the devices or entities.
Second, it will also have to manage the external trust management information. This can be realised by connecting to the trust broker to check for other identities, devices, enterprises and their relationships. This means that it will handle the external identities, devices and enterprise relationships.
The identities of the devices and the employees will be handled as the Identity Management Services does by using the Identity Lifecycle again. However, instead of (de-)provisioning one will have to on-board and off-board the identity and ensure that one will work with a PII broker in order to handle the privacy information.
The enterprise relationships will be handled differently. One will have to check the trust state, capabilities, competencies and reputation of the employees, devices and the complete enterprise. Furthermore, the service will collaborate with the information asset management services to see what assets are exchanged. It will also work together with the Enterprise Relationship Lifecycle Management processes to exchange the necessary information with the trust management framework in order to execute these processes.
Third, it can also distribute, access, allow creation of the legal/regulatory/compliancy/ assurance artefacts by connecting to the trust broker where these artefacts will reside. This will be very important to support several penalty systems, enforcement mechanisms and the processes to establish contracts and check for them.
Furthermore, the service will allow the exchange of information in many processes such as the Person-, Device-, Information-, and Enterprise Relationship Management processes.
The service will allow one to create new trust relationships with other entities such as devices, employees or complete organisations by connecting to the trust management framework and establish contractual means in order to be allowed to check for the trust state of these entities. However, the exchange of all of the collaborative data will be done in cooperation with the information asset management service.
Finally yet importantly, the service will log all of the actions that have been undertaken by the services for accounting.

However, the trust broker itself is of the scope of this service. The service will only have to connect to the trust broker and use a trust management framework to manage the trust level. The details of the trust broker and the framework themselves can be found in section 2.6.8. The PII broker will remain of scope as well. All the service will do is connecting to the PII broker to ensure that the subject of the PII will be in control of his own data as soon as any PII data will have to be handled. See for more details section 2.6.6.

Description – Classification:
The next group of services is focused on data classification. The group will act as one service and classify the information assets as soon as they are created, updated or incoming as a new asset from an outer organisation. It will also reclassify the same assets if necessary.

However, since there is a large discussion going on whether the classification should be automated or not (see section 2.6.6 for more details), one could argue that both sides should be heard and that the service should allow automated, non-automated and computer assisted classification. So multiple services should be implemented that allow different ways of classifying the information assets. The different automated services should also allow different approaches of automated classification, with different algorithms, hardware et cetera.
It would be highly desirable that all of these services should output in the same classification model. However, as we have already seen with the laws of identity, it could also become quite harsh to introduce a common classification scheme as well and ensure that all entities in the collaborative environment will hold to them. So in order to ensure that one can communicate his classification terms, the service should have several interpretation services that will allow one to communicate the classification level by several terms such as "red data" (traffic light protocol, see section 2.6.6) or "top secret" (used term for really important secret information assets). [128] The service should also support the Risk Management processes in order to allow one to define the information risk within the asset in the terms of the risk taxonomy.
The service should not give a static outcome of the classification processes, it should allow reclassification on differential rates, according to the asset that is being classified. This should allow dynamic reclassification and thus temporal classification of several assets. This means that either the service will be triggered by several of the COA framework Processes or Services in order to reclassify the information asset, or it should be triggered by itself based on certain timers to classify the information asset again.
Another important aspect of the service will be the detection of PII. As soon as PII is detected in the to be classified information asset, it will immediately cooperate with the trust management service that resides in the same complete service set (or later in a separated implementation) in order to contact the PII broker and ensure that the subject will be notified of the new PII that will be used about him, allowing him to manage his own PII.
Getting back to those services, the classification will have to happen in collaboration with several other services such as: Identity Management for establishing on which device and by who it is made, Risk Management processes such as the risk assessment process to see what kind of risk the information asset will carry along with itself. Enterprise Relationship Management processes and trust management services in order to check the classification state of the incoming information assets.[129] The Information Asset Management service that will allow one to check the metadata of the information assets and check if a reclassification is necessary. The same service will be used to take certain information risk lowering measures depending on the outcome of the classification.
The service will also have to reclassify the information as soon as a part of the asset has been updated. This means that it will have to be triggered by the Information Lifecycle Management processes as soon as this happens.

Finally yet importantly, the service will log all of the actions that have been undertaken by the services for accounting.

Requirements – Trust management:
The following requirements can be derived from the findings of chapter 2 and 3:

- General requirements:
  - The service should be capable of accessing legal frameworks in order to use the country/legal specific procedures for the complete Lifecycle Management processes .

[128] One could argue that this is not necessary if everyone in the collaborative environment will have adopted the COA framework. However, as long as that is not the case, one will certainly have to use several classification terms.

[129] One could also argue that this is obsolete, if the classification of het information asset is recorded in the metadata of the information asset.

- The processes should support oncoming trust management standards and protocols, and their respective repositories for reputation, digital identity, contractual information et cetera.
- The services should be capable of using an ontological structure to combine trust and reputational values.
- The service should be capable of weighting the importance of several reputational facts in order to get to a judgement of the reputation and trustworthiness of an entity.
- The service should be capable of using the outcomes of a risk assessment and risk reducing acts for the trustworthiness checks.
- The service should re-evaluate the reputation and trustworthiness of all entities on a frequent base.
- The service should be capable of deciding the actual reputational values and trustworthiness of another entity based on several outputs.
- The procedures for checking the several sources and the actual calculation of trustworthiness and reputational values should all be implemented separately.
- The service should be capable of handling the results of behaviour and obligation monitoring.
- The service should have a separate set of services that work together with the audit services in order to monitor the behaviour of the entities in the collaborative environment.
- The service should have copies of the trust management framework repositories for reputation, identities et cetera.
- The service should be capable of using several definitions of trust and risk such as the risk taxonomy.
- The service should be capable of handling several legal frameworks to have the capability of interpreting the legal information provided by the trust broker.
- The service should be capable of notifying all of the services/processes (especially the audit service) of new legal information updates based on new collaborations.
- The service should be capable of communicating with several types of trust brokers, versions of the trust management framework and the PII brokers.
- The service should be capable of handling dynamic amounts of information based on the relationship and the asset at risk (JFC1).
- The services should be implemented in such a way that they provide enough flexibility to grow as more standards and protocols will be developed.
- The services should be simple to understand, scalable and easy to manage (JFC2).
- The services should be based on open and secure protocols and standards (JFC4).
- The services should have the capability to deliver a transparent level of trust of all of the devices, identities and other assets, including the service itself (JFC 6, 7).
- The service should have a high availability and a consistent high performance.
- The services should log all of the actions for auditing reasons.
- The services should take the Privacy concerns in mind, even when communicating with the PII broker et cetera.

- Requirements for internal trust management, which is the management of the trust/reputation status of personnel and devices:
  - All personnel should be registered at the trust broker by the People Lifecycle Management processes.
  - The services should allow personnel to manage their PII via the trust broker that is contacted by this service. This includes the PII that is managed by the Identity Management services.
  - The reputational status of the personnel will be accessible by personnel itself.
  - The reputational status of the personnel will have to be checked by the audit service and checked against audit results.
  - All devices should be capable of connecting to the trust management framework and the trust broker.

  - All devices should be registered at a trust broker by the Device Lifecycle Management processes.
  - The service should be capable of checking the reasons and processes behind the reputational and trustworthiness outcomes.
  - The trust state of the devices should be communicated to the trust broker (e.g. the end-point security status and the device his capabilities and environment) .
  - The trust state of the devices should also be read from the trust broker. Depending on the implementation of the end-point security service.
- Requirements for external trust, which is about the exchange of trust and reputational data between the trust broker and the service:
  - The service should be capable of checking the current organisational relationships and map them in order to understand the influence on the reputational data.
  - The service should be capable of checking the trustworthiness and accuracy of the data that is provided by the trust broker by examining it and comparing it to other data such as their own accountability data about themselves and other organisations.
  - The service should be capable of exchanging reputational information with other parties outside the trust broker for a trust broker trustworthiness and accuracy test.
  - The service should be capable of checking the trustworthiness and the reputation of all of the entities in the collaborative environment based on its own accountability data.
  - The service should be capable of checking the reasons and processes behind the reputational and trustworthiness outcomes.
  - The service should have the capability to verify the reputations of the other organisations.
  - The service should have the capability to manage the contracts between the different parties in collaboration with the other COA framework Services.
- Requirements for handling external identities of devices and personnel:
  - The service should be capable of handling the Identity Management parts for external devices and personnel. This means that identification, on-boarding, authentication, management, off-boarding and other means will have to be processed by this service for all external devices and personnel.
  - The service should have the capability to verify the identity of the other organisations.
  - The service should be capable of requesting and checking the trust status of external devices (e.g. security status, capabilities et cetera).

One should notice that the list of requirements is not exactly complete. This is because of the complexity of the service. More research will be necessary on this field.

Requirements – Classification:

The following requirements can be derived from the findings of chapter 2 and 3:

- The service should allow multiple classification models.
- Each classification model should be implemented by a separated service.
- Each classification model should be translatable into another by a dedicated service.
- The service should allow multiple classification methods and algorithms, either automated, computer aided or non-automated.
- Each algorithm should be implemented in a separated service.
- The service needs to be capable of handling all of the available data containers, from XML messages to RDBMS-cells, from Portable Document Format files to Bitmap files.
- The service should be capable of using multiple algorithms by using several services either parallel or in series.
- The service should be capable of classifying the information asset based on the context of the asset. The services should consider several aspects such as the topic, surrounding data,

security policies, compliancy policies, date/time that it is build, surrounding documents et cetera.[130]

- The service should be able to see whether the information asset should be re-classified on a later moment and report, if it suspects to do so, to the Information Asset Management Service.
- The service should be able to recognise and classify PII. It should report its findings to the Information Asset Management Service so it can manage it properly. It should also report to the Trust management services in order to report to the PII broker of the (newly) found PII.
- The service should be capable of classifying the risk according to the risk taxonomy.
- The classification services should be reachable by all of the processes and services that create new information to make sure that it can be classified by the service.
- The service should be capable of incorporating the results of the classification in the metadata of the information asset.
- All of the actions that are undertaken by the services should be logged for accountability.
- New incoming information assets should be checked for classification data. If the information is absent, then the service should classify the information.
- The services should be capable of partial classifying assets, so that the asset can have several classification levels for different parts of the asset allowing one to check those parts of the assets for which the classification level is not too high.
- The classification services should be simple, scalable and easy to manage (JFC2).
- The services should be both reachable in on- and off-line mode so that classification of the asset will always be possible.
- The services should be capable of being executed in multiple environments and on multiple devices (JFC 3&5)
- The services should work based on open and inherently secure protocols (JFC 4).
- The services should maintain confidentiality and the integrity of the information assets during their processing.
- The services should work efficient and with a high availability.
- All of the actions of the classification services should be recorded for auditing.

One should notice that the list of requirements is not exactly complete. This is because of the complexity of the service. More research will be necessary on this field.

Recommendations for both Trust management and Classification:
The following requirements can be derived from the findings of chapter 2 and 3:

- **Separate services:** in order to make the services more manageable, one should separate the classification service from the trust management services. Both of the services are focussed on different processes and aspects. This will remove some of the complexity and allow a more dynamic implementation of the services, since they will not be related so tightly anymore.
- **Separate End-point security service:** the End-point Security service should be separated from the Trust Management services in order to reduce the complexity of the trust management services and make it more manageable.
- **Use current existing technologies:** to ensure that the services named in this section will be workable in the current situation, one should try to use currently existing technologies to implement them. This should allow a workable system.
- **Services implemented in a Service Oriented fashion with a layered process approach:** to ensure that one can use several types of protocols, classification algorithms, trust broker mechanisms, end-point security mechanisms, standards et cetera, one should implement the

[130] See section 2.6.6 for an enumeration of aspects which should be considered.

services loosely coupled with clear in- and output definitions per service in order to ensure interoperability between the services and thus the Identity Management (sub)systems.

- **Roadmap – prepare for trust broker:** there should be a roadmap for the trust management services, which focuses on working with them to be developed into a trust management framework. It should first allow P2P interaction and federation, and later on work with a trust management framework and a trust broker. One should ensure that there is enough flexibility in the current solution and the solutions to come, so the trust management framework and PII broker can be implemented.
- **Roadmap – prepare for multiple types of classification:** there should be a roadmap for enhancing the classification services, starting with non-automated classification, building towards advanced computer classification mechanisms that allow one to classify the ever growing amounts of information that is being created, updated and exchanged on a daily basis.
- **Roadmap – prepare for end-point security for every device:** The end-point security service should no longer be focussed on an end-point. It should comprise all of the devices in order to give full transparency to the risks that are present in a collaborative environment.
- **Further research:** the service and the available/required technologies should be further researched in order to make it a workable set of services that can be implemented conform the requirements and recommendations.
- **Cross-COA-service research:** one should research what data should exactly be exchanged between the services inside the Identity Management service and between all of the COA framework services.
- **Create a set of inherently secure open standards:** to ensure that all of the implementations along the roadmaps will be universally exchangeable, one should create a set of inherently secure and open standards, which will allow one to follow all of the requirements and recommendations written in this document.

COA V2.0 UPDATES – part one:

The following relevant additions have been made in the COA 2.0 revision:

- **A global overview of what is necessary for trust management**, in: "Trust Management – A Brief Overview", which provides the following additional requirements and service descriptions:
  - *Prevention against Doomsday scenario:* The usage of the trust management service provides prevention against the doomsday scenario of the "Loss of confidence".
  - *The emphasized need for interoperability:* The new revision emphasizes interoperability between standards to make this component usable as a replacement for the current inherently insecure methodologies.
- **An additional classification scheme for** , in: "COA Service Trust Management: Impact Sensitivity Categorization", which provides the following additional recommendations:
  - *The need for a common language (taxonomy):* there is a need for a taxonomy and a set of trust levels defining impact sensitivity of information, based on confidentiality, integrity, availability and authenticity.
  - *The usage of six levels for business impact classification:* 6 levels have been defined: catastrophic, material, major, minor, insignificant and none. The exact ranges for the financial damange values related to these schales will still have to be adjusted to suit each organization.
- **Additional details around Data classification**, in: "COA Paper Information Classification", which provide the following additional recommendations and descriptions:
  - *The existing lists of classification standards:* there is already a lists of relevant classification standards, which can be found at http://xml.coverpages.org/classification.xml .
  - *A better definition of PII:* the EU directive 95/46/EC has been taken into account in Rev2.0 of the COA framework. See the Position Paper for more details.
  - *Issues around consistency.* The Position Paper shows that it will be hard to apply consistent information classification. This should be taken in mind while using the service.
  - *The use of automated classifications.* In Rev 2.0 there is more space for automated classification. See the Position Paper for more details.
  - *The use of multiple classifications:* one should use multiple classifications.
  - *The impact of data aggregation:* data aggregation will impact the classification levels. See the position paper for more details.

COA V2.0 UPDATES – part two:

- **The new basic parts of trust management**, in "Position Paper – COA Framework", show a new outline of the trust management, which can be summarized as follows:
    - *Business impact levels*: there are five levels of impact proposed: catastrophic, material, major, minor, insignificant. See also the next major bullet.
    - *Information classification:* the information classification should be included, whereas they use the traffic light protocol.
    - *Impact sensitivity categorization*: there should be an impact sensitivity categorization of the information based on measures of it's confidentintiality, integrity, authenticity & availability, whereas the same five levels of the business impact levels should be used, with the additional level "none".
    - *Control Stratification:* a set of standardized information trust categories by trust level would be required. One could define a six-level trust taxonomy for authenticity: Assured, affirmed, proven, confirmed, asserted and unknown. See the Position Paper for more details.[1]
    - *Architecutre Segmentation Model:* a coherent architectural mdoel is required to map the Trust Management components into an effective operationally aligned structure.
- **The introduction of a common language for Business Impact**, in "Trust Management: Business Impact – A Common Language", which can be summarized as follows. The position paper shows that communicating the implications of a risk in terms of the potential business impact has become vital. That is why they introduce a set of impact definitions for the impact on four domains (human life, financial, brand and environmental impact) with the following impact levels:
    - Disastrous Significant loss of life, Collapse of multiple enterprises or a countries economy, significant global environmental incident
    - Catastrophic Loss of multiple lives, Significant financial loss, Collapse of an enterprise, Significant countrywide environmental incident.
    - Material Accidental loss of life, Financial loss of reportable sums of money, Significant brand impact, Significant local environmental incident.
    - Major Significant Injury, Significant financial loss, Brand impact, Local environmental incident
    - Minor Injury, Financial loss, Local Brand Impact, Minor environmental incident
    - Insignificant Negligible injury, Slight financial loss, Negligible environmental impact

  These levels mandate controls to be enforced that protect the information protection requirements in terms of Confidentiality, integrity, availability and authenticity.
  The paper also identified the need for a Business Impact Scale which should have sufficient granularity and a clear and unserstandable set of definitions of business impact.
- **The usage of Control Stratification**, in "COA Service Trust Management- Control Stratification, which can be summarized as follows:
    - *As an addition to the classification scheme and the impact sensitivity categorization:* these two drive control requirements to ensure the protection of information in de-perimeterised environments. One of the control requirements is to establish a level of trust in the identity of entities that access and handle information. Control Stratification enables trust in an identity to different levels based on the level of authentication given by an entity.

COA V2.0 UPDATES – part three:

- *The six levels for authenticity.* The six levels that have been specified consist of:
  - C5: ASSURED (biometric)
  - C4: AFFIRMED (positive physical or logical authentication)
  - C3: PROVEN (authenticated by trusted third party)
  - C2: CONFIRMED (confirmed by strong attributes)
  - C1: ASSERTED (self-asserted)
  - C0: UNKNOWN (no authenticity assertions made - anonymous)

See the Position Paper for more details.

The publications can be found at: https://www.opengroup.org/jericho/publications.htm
[1]: as this is part of the Trust Management service and related to policy management, we decided to shortly note this in the Rev 2.0 update of section 3.7.4 and here as well.

3.7.4. Policy Management

Introduction
This service is focussed on managing the policies. That means that the creation, negotiation, administration, information providing, exchanging and monitoring will be done by this service. All kind of policies will have to be managed by this service, yet information access policies will be the main focus for now.
The description, requirements and recommendations in this section will be based on sections 2.6.5 (Policy management), 2.6.6 (Information access policies), 2.5.4 (Roles and relations in collaboration) and paragraphs 3.3 (COA framework overview) 3.5 (COA Principles), 3.6 (COA Processes). See also the named paragraphs and sections for more details.

Description:
This group of services is focussed on the different aspects of policy management . In (Forum 2007c) the focus was set on managing the information access policies. However, one could argue that there are more policies to cover in a collaborative environment such as specific corporate policies for behaviour, usage of assets, security policies, policies derived from good corporate citizenship et cetera.

Let us try to give a more organised description of the service then:
The policy management service will exist of several services that will execute certain important processes such as:

- **Policy negotiation:** there are various moments when two or more parties will have to negotiate which policies should be used in the collaborative environment. This can vary from how to deal with customers to how information will have to be exchanged.
- **Local policy management:** the policies will have to be managed locally (on- and off-line) on each device. There are many policies to be managed, one could think of information access policies, security policies et cetera.
- **Policy creation/administration:** there are various moments when policies will have to be created such as when one enters a new collaborative relationship (business policies), or when one creates a new information asset (information access policies).
- **Providing information for policy decisions:** there are various moments when one will request to be allowed to execute a certain action. The policy decision process will check whether one may execute the action based on the active policies that are related to the action. Access to a file is a good example. One will have to ask if he is allowed to access the file at the Policy Decision Point (PDP). The PDP will check based on the information access policies if the user is allowed for accessing the file. However, the policy decisions are not part of the service. The decisions will be made by other services and entities, which will retrieve information from the policy management service.

- **Policy exchange:** there are various reasons for exchanging policies. One of the most important one is when information assets will be exchanged. Whenever that happens, the information access policies will be conveyed with the assets in the metadata of the assets.
- **(Partial) Policy enforcement:** the service is partially involved in enforcing the policies. It could for instance alert whenever a policy is negated.
- **Policy (process) monitoring:** the service will have to monitor all of the policies and the actions and processes related to these policy processes. This will allow one for instance to check whether the (information access) policies are still in place and actively used.

The policy management services will have to act with many processes such as:

- **Information management and protection:** these processes are part of the information asset management. Whenever an information asset is created, altered or whatsoever it will have to be checked against it current policies, whether those who want to alter the information are allowed to. If so, one will have to check the policies themselves to see whether these need to be altered as well. Whenever a non-public file becomes public after altering or vice a versa, information access policies will have to change as well.
- **Start of a relationship:** Policy management in a collaborative environment already starts when two organisations start a collaborative relationship. The basic policies about that relationship will have to be exchanged in order to manage the relationship on both sides. These could be general information access policies and others.
- **Information exchange:** whenever an information asset is exchanged, the information access policies will have to be exchanged as well.
- **Compliancy processes:** another important interaction will be with compliancy processes. In order to ensure that the collaborating parties will be compliant to security standards, one could choose to exchange security policies to ensure the same interpretation of those standards.
- **Preparation for the information exchange:** before the first information asset is exchanged between two parties, the policies that manage the policy management system itself will have to be exchanged to ensure that the information access policies that need to be exchanged between the two parties alongside with the asset will be properly interpreted.
- **Relationship management:** (information access) policies will have to be exchanged during the relationship management in order to make sure that both parties will have the capabilities towards each other that they have been allowed to by contractual means.
- **Introduction into an environment:** whenever a new party is entering a collaborative environment, then loads of (information access) policies will have to be exchanged between the new party and the collaborative environment to ensure proper information exchange and management of devices and identities.

Finally yet importantly, the service will log all of the actions that have been undertaken by the services for accounting.

Requirements:

The following requirements can be derived from the findings of chapter 2 and 3:

- The service should allow the data owner to set information access policies that describe how his data should be handled.
- The service should ensure that the policies are bound to the information asset and to all of its copies.
- The service should allow multiple governance patterns for policies such as automated control, workflow-based control, accountability and time-limited permissions.

- The service should only provide policy administration and monitoring. It should not provide decision and enforcement.[131]
- The policy management service will have to work together with the other services to ensure the policy distribution.
- The service should be capable of using multiple policy languages for defining ACLs or other information access policies.
- The service should be capable of converting the policy languages to one another and should be capable to translate them to human readable policy information.
- The service should be capable of applying n policies to n assets (JFC1).
- The service should be capable of implementing information access policies to parts of an asset (JFC1).
- The service should be scalable, simple and easy to manage (JFC2).
- The services should be based on inherently secure protocols and standards (JFC4).
- The services should run on any hardware/OS that is used today or in the future (JFC5).
- The information access policies should also be conveyed in DRM so that it is automatically enforced (JFC9).
- The service should not have a negative performance on the systems.
- The service needs to be capable of handling all of the available data containers, from XML messages to RDBMS-cells, from Portable Document Format files to Bitmap files.
- The service should be available at all times in off- and online mode.
- The service should log every action to make it auditable.
- The services should be capable of using other languages for other types of policies as well.
- The services should be easily expandable to manage the policies on any other field.
- The service should be capable to exchange any type of policy with multiple instances of the service inside a collaborative network.
- The service should be capable of communicating and exchanging the information access policies prior to the asset itself.
- The service should be capable of checking the state of the security policies and report its findings to the audit service.

One should notice that the list of requirements is not exactly complete. This is because of the complexity of the service. More research will be necessary on this field.

Recommendations:
The following recommendations can be made on the field of this service:

- **Use current existing technology:** to ensure that the Policy Management service will be workable in the current situation, one should try to use currently existing technologies to implement the services. This should make it a workable system.
- **Implement the solution in an open flexible Service Oriented fashion:** to ensure that one can use several protocols/standards/ policy types, one should implement the services loosely coupled with clear in- and output definitions per service in order to ensure interoperability between the services and thus the policy management (sub)systems.
- **Further research:** the service and the available/required technologies should be further researched in order to make it a workable service that can be implemented conform the requirements and recommendations.
- **Cross-COA-service research:** one should research what data should exactly be exchanged between the services inside the Identity Management service and between all of the COA framework services.

[131] If the services are implemented separately from each other, then one could argue that it is ok to let all of the aspects be covered by the policy management service.

- **Understanding one's position in the collaborative network:** the service should be manageable in such a way that the manager can take the role of the organisation in the collaborative network into account. See section 2.5.4 for more details.
- **Roadmap – prepare for a fine-grained information access policy infrastructure:** there should be a roadmap for policy management that is focussed on implementing a globally accepted, fully accountable, very fine-grained information access policy infrastructure for policy management. One should ensure that there is also enough flexibility in the current solution to implement such an infrastructure.
- **Roadmap – prepare for a service that covers most types of policies:** there should be a roadmap for policy management, that focuses on implementing a globally accepted, policy management service that covers any of the fields where digital policies are used. One should ensure that there is also enough flexibility in the current solution to implement such a service.
- **Create a set of inherently secure open standards:** to ensure that all of the implementations along the roadmap will be universally exchangeable, one should create a set of inherently secure and open standards that will allow one to follow all of the requirements and recommendations written in this document.

COA V2.0 UPDATES:

The following relevant additions have been made in the COA 2.0 revision:

- **The usage of access control in context of information lifecycle management**, in " COA Paper Information Lifecycle Management", which can be summarized as follows:
  - *The usage of authentication and authorization:* these should be applied to principles requesting access to information. A set of control stratification levels should then be used to define the correct strength of authentication.
  - *Appropriate access controls.* The access controls should be chosen appropriately, reflecting the security requirements defined in the Information Classification and Impact Sensityvity Categorization stages from the Information Lifecycle Management process.
- **The usage of Control Stratification**, in "COA Service Trust Management- Control Stratification[1], which can be summarized as follows:
  - *Control Stratification:* a set of standardized information trust categories by trust level would be required. One could define a six-level trust taxonomy for authenticity: Assured, affirmed, proven, confirmed, asserted and unknown. See the Position Paper for more details.

The publications can be found at: https://www.opengroup.org/jericho/publications.htm

[1]: see also the rev 2.0 update at section 3.7.3 for a more complete coverage

3.7.5. Meta/Information Management

Introduction

The Meta/Information Management or "Information Asset Management" service is the service that will manage all of the information assets. It will manage all of the information assets together with the Trust Management and Classification and the Policy Management services.
The description, requirements and recommendations in this section will be based on section 2.6.6 (classification, protection and privacy issues) and paragraphs 3.3 (COA framework overview) 3.5 (COA Principles), 3.6 (COA Processes). See also the named paragraphs and section for more details.

Description:

This service or set of services is focused on managing the lifecycle of the information assets. This means that it will ensure that the data is appropriately secured and managed by its services from creation, storage, transit, use (altering/access/reading) and destruction. It will

also take the policy decisions for the information access policies that are added to the information assets based on the classification. This means that the service will check whether with the policy management service if an action requested by a principal against a certain information asset will be allowed or denied.
There will be several services that will cover for different aspects of the information lifecycle such as:[132]

- **Information in storage:** there will be a service set that will ensure that the information will be properly secured by applying the right type of encryption, cipher strength and DRM related mechanisms. It will also ensure that the proper information access policies will be added to the assets in such a way that they cannot be easily bypassed. The exact type of encryption, DRM and policies that will be attached are depending on the outcome of the information classification process.[133]
- **Information in transit:** a set of services will have to take care of the information in transit. If one wants to get the information in transit, then there are various aspects to be considered:
  - *Sending:* the first one is the sending of the information asset. The service will have to check whether it is in line with the policies that this principal can send the information asset to another principal. Whenever it is send, the services will have to check where the information is sent to and where it is now. DRM related mechanisms and the Trust Management and Classification –service can help reporting the location of the information asset to these services.
  - *Communications*: the second one is communications. The classification level should dictate what kind of security should be applied to the communications (e.g. the SSL attributes that will have to be enabled/set et cetera). Several services inside the information asset management should collaborate with other services and processes (e.g. Trust Management and Classification and a separated end-point security service) to see whether the device that will check the information has the capability of setting up secure communications to either send or receive the asset.
  - *Spreading*: the spreading of the information assets will be monitored and managed as well. Spreading is interpreted here as sending to multiple recipients. Again, the sending process will be repeated, yet interpreted as a spreading process with the same associated information management services. The services will first check whether the principal is allowed to spread the information across that medium to those other principals and then the assets will be tracked.
- **Use of Information:** the information assets can be used in several ways. One can access, read, update the information or copy or cut it into other assets. Notice that there is a difference between accessing and reading. One might be allowed to access a certain information asset container, yet not read the entire contents of that container.
  - *Access:* The service will have to check whether it is in line with the policies that this principal can access the information asset with the device he is using and the related trust states. If not, access is prohibited and otherwise granted.
  - *Read:* The service will have to check whether it is in line with the policies that this principal can read the information he wants to read, if not, the information will be blanked out otherwise, he can read it.
  - *Update:* The service will have to check whether it is in line with the policies that this principal can update this part of the information asset. If not, it will not have the capability of doing so and otherwise it will be granted.

[132] One should never forget that all of these services will only be activated in a certain form if the classification level and the related policies that come with that classification level are in need of these services.

[133] The backup services are of the scope of this service for now.

  - *Copying and cutting:* The service will have to check whether it is in line with the policies that this principal can either copy or cut information of the asset and paste it into another container. If not, the actions will be unavailable, otherwise he will be able to do so.
- **Information destruction**: the information assets could have to be destroyed as well. The service will take care of this as well. It will destroy the information asset and all of its copies (if necessary).
- **Meta-data management:** The meta-data of the information assets will have to be managed as well by these services. Which means that several actions should either be taken care of by the services or supported by them such as:
  - *Creation:* the metadata and DRM will have to be created based on the outcome of the classification process. The data will have to be incorporated in the body of the information asset.
  - *Addition:* If any information is added to the asset, then new meta-data might have to be added as well. This will again be taken care of by these services.
  - *Reading:* the reading of the meta-data might be taken care of either by the service or by tools that allow accessing the information asset container.
  - *Updating:* If the asset is updated, or its classification has changed or if the policies surrounding it have changed, then the meta-data will have to be updated by these services.
  - *Destruction:* if (a part of) the information asset container is destroyed, then the meta-information might have to be destroyed as well. This will also be done by these services.
- **PII Management:** another important process, which is tightly related to the information lifecycle, is the management of PII. This service will have to (partially) take care of it. As we have already seen in section 3.7.3, it will be managed by a PII broker. Well, the interconnection with the subject will be managed with the PII broker. The information itself will have to be managed by a set of coordinated services that will connect to the PII broker through Trust Management and Classification. These services reside in the Information Asset Management Services. They will make sure that the information is properly secured as any other information asset, yet with the issues of privacy taken in mind as already discussed in previous sections, paragraphs and chapters.
- **Policy decision:** Finally yet importantly is the policy decision service. As it is not a part of the information lifecycle itself, it is an important service that is actually already referred to in several of the aspects of the information lifecycle processes that have been named in the context of this service. This service will either allow or disallow the actions that have been named here, based on the principal that is requesting the action and the policies surrounding the asset to which the action should be applied.

Finally yet importantly, the service will log all of the actions that have been undertaken by the services for accounting.

Requirements:
The following requirements can be derived from the findings of chapter 2 and 3:

- The service will have to be able to apply multiple types of encryption in both terms of algorithms and strength of the key.
- The service will have to be capable of handling multiple types of Digital Right Management protocols.
- The service should always be available, either on- or offline.
- The service should be capable of interpreting the outcomes of the classification process.
- The service either needs to support multiple classification models or it needs to cooperate with the information classification service to interpret the classification levels.
- The service needs to be capable of handling all of the available data containers, from XML messages to RDBMS-cells, from Portable Document Format files to Bitmap files.
- The service should be capable of tracking back the location of an information asset and its copies at all times.

- The service should be integrated into or cooperate with all of the applications that handle information assets.
- The service should be able to check whenever one wants to send the information by any message or when one wants to spread the information asset by the internet or multiple physical mediums.
- The service should be capable of interpreting multiple security protocols for network traffic and for sending the information.
- The service needs to be capable of cooperating with all of the other services and processes inside the COA framework and manage the information that come from these entities.
- The service needs to be capable of blocking the capability of copying or cutting information from an information asset.
- The service needs to be capable of partial blanking out information assets.
- The service needs to be capable of destroying an information asset and all of its copies.
- The service needs to log all of its actions for the auditing service.
- The service needs to be capable of handling PII in cooperation with the PII broker via the trust management services.
- The service needs to be capable of executing the correct policy decision at all times.
- The service should protect its own data repositories and classify them in cooperation with the classification services.
- The service needs to be capable of following the different legal and regulatory laws on the field of information protection so that it can be used anywhere.
- The service should be capable of handling data outside of the user his control by working via the trust broker or in a P2P fashion with the DRM in the metadata of the information assets.
- The service and the DRM that it applies should be capable of rendering an asset completely unusable as soon as it falls into the hands of those that are not allowed to access it.
- The service should be capable of working on a fine-grained and a coarse-grained information infrastructure.
- The service should not rely on network connections.
- The service should be capable of managing keys in such a way that it can work in both on- and offline mode and allow one to easily add/revoke/change a key.
- The service should control all data in any state, at any given time at any place.
- The service needs to be capable of working with any kind of identity in cooperation with the Identity Management service and the trust management service in order to be capable of (de-)authorising any principal to any information asset.
- The service should be based on inherently secure open standards and protocols.
- The service should be capable of executing all of the actions that are necessary for the Information Lifecycle Management.[134]
- The services need to be simple, scalable and easy to manage (JFC2)[135].
- The service should not have a negative impact on the availability of the information assets other than the rightful denying access to assets due to the absence of the correct authorisations.
- The service needs to be effective and efficient at all times by working in both off- and online mode without having a negative impact on the performance of the information systems.

One should notice that the list of requirements is not exactly complete. This is because of the complexity of the service. More research will be necessary on this field.

Recommendations:

[134] Of course, the classification, Identity Management related actions et cetera, are not part of the actions mentioned here.

[135] The other JFCs are already built into the requirements, without notice.

The following recommendations can be made, derived from the findings of chapter 2 and 3:

- **Use current existing technology:** to ensure that the Information Asset Management service will be workable in the current situation, one should try to use currently existing technologies to implement the services. This should make it a workable system.
- **Implement the solution in an open flexible Service Oriented fashion:** to ensure that one can use several protocols/standards/ types of classification / types of DRM / types of information asset containers / applications and devices, one should implement the services loosely coupled with clear in- and output definitions per service in order to ensure interoperability between the services and thus the Identity Management (sub)systems.
- **Roadmap - Prepare for a fine-grained information infrastructure:** there should be a roadmap for the information management services that is focussed on implementing a globally accepted, fully accountable, fine-grained information infrastructure. One should ensure that there is enough flexibility in the current solution to implement such an infrastructure.
- **Further research:** the service and the available/required technologies should be further researched in order to make it a workable service that can be implemented conform the requirements and recommendations.
- **Cross-COA-service research:** one should research what data should exactly be exchanged between the services inside the Identity Management service and between all of the COA framework services.
- **Create a set of inherently secure open standards:** to ensure that all of the implementations along the roadmap will be universally exchangeable, one should create a set of inherently secure and open standards, which will allow one to follow all of the requirements and recommendations written in this document.

COA V2.0 UPDATES:

The following relevant additions have been made in the COA 2.0 revision:

- **Additional EIPC needs**, in: "COA Paper Secure Data: Enterprise Information Protection & Control", which can be summarized as follows:
  - *A standard for handling in-clear classification information:* EIP&C systems must have enough classification information in-clear to ensure that non EIP&C systems understand how to correctly handle that document. The information needs to be stored in such a way that tampering with that in-clear information will be detectable.
  - *The need for an inherently secure protocol for communicating protected data* between the consumers of EIP&C and the server or enterprise that controls the data's EIP&C attributes.
- **Structural changes**, in: "Position Paper – COA Framework", which can be summarized as follows:
  - *Information asset management refered to as "Information Taxonomy and semantics":* Eventhough the process is renamed, it still covers the same aspects.

The publications can be found at: https://www.opengroup.org/jericho/publications.htm

3.7.6. Audit

Introduction

The last service to discuss, is the audit service. This service, again consisting of a set of services will audit all of the processes and services that have been named inside the COA framework.
The description, requirements and recommendations in this section will be based on section 2.6.10 (Audit) and paragraphs 2.4 (Control Objectives for Information and related Technology),

3.3 (COA framework overview) 3.5 (COA Principles), 3.6 (COA Processes). See also the named paragraphs and section for more details.

Description:
This service will audit all of the named services and processes by auditing the auditable events such as information storage, transfers and retrievals. It will also audit the business controls associated with each service and process. Both the availability and implementation, as well as the status of the control will be audited by the service.
The audit service will consist of several services that will do several types of audits[136], varying from operational to compliancy audits, from administrative to information security audits. Each type of audit will have its own specialised, standardised set of services. Additionally, services will be developed for audits focussed on contractual means.
The service will have to do periodical audits of all of the parties in the collaborative environment with their respective processes and services.
Some services will be specialised in executing spot audits or stealth audits that will not be noticeable by the party that is being audited.
The audits themselves will differ from audits as we know them. The scope, the amount of controls and the amount of separate services will increase over time, as companies will de-perimeterise safely using the COA framework.
This means that the audit will have to be optimised for a new tactical and strategic scope. Meaning that one will have to use revised versions of the current framework that will allow one to conduct the audits in an appropriate way.
The result of the audits will be often sent to the trust broker and used for reputational, legal, regulatory and contractual means. This allows the collaborative environment to use the results as a partial assurance and as an input for the Risk Management processes.

Requirements:
The following requirements can be derived from the findings of chapter 2 and 3:

- The audit service will have to be capable to work with several governance frameworks and measures (as defined in section 2.4.2) in order to execute the several types of audits.
- The audit services itself will have to log all of its actions for Risk Management and further audits.
- The service should be able to handle all kinds of metrics in order to ensure that the principles and quality attributes as well as other, yet non-defined, attributes will be measurable by the audit service.
- The service should be capable of handling several kinds of benchmarking methods in order to test the performance and the capability of both (non-) COA framework entities.
- The service should be capable of cooperating with the Risk Management processes in order to create more transparency and check the risk assessment outcomes.
- The service should be capable of auditing brokers such as the PII broker and the trust broker.
- The audit process should be covered by a set of services that allows efficient audit planning, execution, evaluation and storage of the findings.
- The service should be capable of processing all kinds of activity goals and languages in which they can be expressed.
- The service should be capable of translating and explaining the meaning of the legal, regulatory, contractual and compliancy means to different languages and objectives, making the service suitable for international collaborative audits.
- The service should collaborate with the Identity Management and trust management services in order to be capable of checking the responsibilities, accountabilities and reputation of all of the identities (both personnel and devices).

[136] One should understand that all of the audits will be done by auditors and might be assisted by technology, yet not executed by the technology itself.

- The service in total needs to be capable of handling any kind of scenario with any kind of sample of control points, varying in scope from the complete collaborative environment, to the processes executed by a single identity. In other words, the service should be completely scalable.
- The service should be capable of presenting the results in a -for both human and machine understandable- report, meaning that the findings should be published in multiple languages and codes. This will allow one to create evidence and means to assure all parties of a contract are complying to the contract or any other compliancy, legal or regulatory means.
- The scope and intensity of the audit should be specific and appropriate to the goal of the audit and the asset at risk (JFC1).
- The services should be simple and easy to manage (JFC2).
- The services should take the context of the audit in mind when the audit is being executed (JFC3). This means the there should be a set of services that check the context of the controls and processes to be audited.
- The audits should not have any negative impact on the availability of the information assets.
- The audits should be performed efficient and effective.

One should notice that the list of requirements is not exactly complete. This is because of the complexity of the service. More research will be necessary on this field.

Recommendations:
The following recommendations can be made, derived from the findings of chapter 2 and 3:

- **Create a set of transparent services that allow efficient computer aided audits:** as audits will change in scope and amount of controls, one will have to create a set of services that are still executed by a human auditor, yet aided by computer systems that will allow an efficient audit.
- **Implement the services in a service-oriented fashion:** in order to provide full scalability and a large scale of audit types and output definitions, the audit service should be implemented in loosely coupled services with clear in- and output definitions per service in order to ensure interoperability between them.
- **Roadmap - Prepare for a new tactical and strategic scope of the audits:** there should be a roadmap for the audit services that is focussed on implementing a globally accepted, fully de-perimeterisation proof audit service.
- **Further research:** the service should be further researched in order to make it a workable service that can be implemented conform the requirements and recommendations.
- **Cross-COA-service research:** one should research what data should exactly be exchanged between the services inside the Identity Management service and between all of the COA framework services.
- **Create a set of inherently secure open standards:** to ensure that all of the implementations along the roadmap will be universally exchangeable, one should create a set of inherently secure and open standards, which will allow one to follow all of the requirements and recommendations written in this document.

COA V2.0 UPDATES:

The following relevant requirements and recommendations have been made in the COA 2.0 revision:

- **Additional findings on the field of IT-audit**, in: "IT Audit and Compliance", which can be summarized as follows:
  - *Requirement: The impact of de-perimeterisation on TOD(Test of Design) and TOE(Test of Effectiveness):* one will need the ability to demonstrate the same risk-based control quality in a de-perimeterised environment as in a bounded one.
  - *Elements that should be in scope of the audit*: the following elements have been named: authentication & authorization services, time stamping, monitoring and auditing, encryption, end point security policies, application security controls (i.e. workflow related), security at entry points (i.e. VPN's, remote users, etc.), third party communications, trust relationships, data centre controls/SAS 70, management of outsourced providers.

The publications can be found at: https://www.opengroup.org/jericho/publications.htm

3.7.7. Summary: the services defined in the COA framework

Looking back at this paragraph, we can coarsely summarise our findings around the services defined in the COA framework as follows:

- **Identity Management, Federation and Reputation** consists of a set of services that will manage the complete Identity Lifecycle of both devices and personnel inside an organisation. From provisioning to de-provisioning and every step in between. We have made a set of requirements and recommendations that are focussed on a systematic approach for using an evolving service that will allow one to take full control over its – user centric - identities over time.
- **Trust Management and Classification** consists of a set of services that will take care of managing the external devices, personnel, end-point security, the relation and the links to the trust management and the PII broker and that will classify the information assets. The requirements and recommendations stated around this service are aimed at creating three separated services. The first one will be the end-point security service that will have to be further developed in order to let it manage all of the devices. The second will be the trust management service which should be developed to a service that allows one to manage the external personnel and devices, the trustworthiness/reputation of all entities and collaborative shared data (in cooperation with the information asset management service) by connecting to the trust management framework and the PII broker. The third service should be the information classification service that should be developed in order to be capable of using several classification models and methodologies.
- **Policy Management** consists of a set of services that will provide information about the (information access) policies and administer them to the assets based on the outcomes of the information classification. It will also exchange and manage policies in their respective expressing languages. The requirements and recommendations made in this paragraph are aimed at creating a policy management service which will be capable of handling all kind of policies and allowing them to be readable to both humans and computers.
- **Meta/information Management** is a set of services that should manage all of the information assets and their respective copies, both on- and offline. It should manage all the actions acted on the asset and ensure its protection during its complete lifecycle based on the outcomes of the information classification and the information access policies that have been applied to it. Requirements and recommendations have been made in this paragraph

that are aimed at creating a service which can manage all of the information assets at any place and any time, piece by piece, using a very fine-grained information infrastructure.

- **Audit** is a set of services that is aimed at conducting several types of audits with several frequencies and scopes, varying from auditing one of the processes of the COA framework to the complete collaborative environment. We have made a set of recommendations and requirements that are focussed on creating an audit service, which will be capable of performing the audits with the new tactical and strategic scope that comes with de-perimeterisation.

3.8.COA quality attributes

3.8.1.Introduction

Attributes of the Solution:
- Usability/Manageability
- Confidentiality
- Integrity
- Availability
- Efficiency/Performance
- Effectiveness
- Agility

Figure 62: Attributes of the Solution (part of the COA framework).

As stated in the beginning of this chapter, after discussing the COA framework in a broader view(section 3.3.2) we will now try to detail the COA framework. See also Figure 47 in section 3.5.1.
The subject of this paragraph will be the COA framework quality attributes. See also Figure 62 for an overview of them.
As already stated in section 3.5.1, the quality attributes can be used to measure if one has achieved its goal. However, what are these attributes? How can one measure them? That is what will be briefly discussed in this paragraph. We will look at each of the quality attributes in section 3.8.2 and summarise our findings in section 3.8.3.

3.8.2.The Quality attributes: description and measurements

The quality attributes:
As one can see at Figure 62, most of these quality attributes are looking familiar. Many of them have been already defined by Stan in (Stan 2008a; Stan 2008b) and summarised in section 2.6.4 as information security attributes (in terms of services and mechanisms).
The quality attributes can be seen as goals or objectives themselves that one can accomplish by successfully implementing the COA framework. This means that the COA framework his purpose is to implement usable, manageable information security measures, which protect the confidentiality, integrity and availability of the information assets. The successful implementation of the COA framework will Furthermore, mean that the security measures will not have a negative impact on the performance of the information system and will still be effective. It will Furthermore, supply a flexible enterprise information architecture.
However, they could also be seen as metrics of a goal that the enterprise management would like to implement themselves from a business perspective such as a safe cost reducing security implementation.
In this section, the quality attributes will be briefly discussed. However, the complete detailed process description of measuring them is out of the scope of this thesis.

Usability/Manageability:
The first quality attribute is derived from JFC2 (“Security mechanisms must be pervasive, simple, scalable and easy to manage”). It is stated in (Forum 2008e) as:

> *“Security measures are non-intrusive, and are easily understood by the individual end-user”.*
>
> (Forum 2008e)

Which means there should be no unnecessary complexity, yet coherency of all the security measurements is an important theme.
Usability can be seen in a bottom-up fashion as that it is about the ease of use of a product, the other one is a top-down fashion that is about the fitness of the product for its purpose: does it what it has to do?[137] As we can derive from the statement in (Forum 2008e) the latter is not important.
Yet, how can one measure the usability and the manageability of the security measures?
The ease of use type of usability can be tested by users themselves, which will have to test them out by using a scenario or realistic situation, wherein the person performs a list of tasks using the product being tested while observers watch and take notes. Several other test instruments such as scripted instructions, paper prototypes, and pre- and post-test questionnaires are also used to gather feedback on the product being tested. For example, to test the attachment function of an e-mail program, a scenario would describe a situation where a person needs to send an e-mail attachment, and ask him or her to undertake this task. The aim is to observe how people function in a realistic manner, so that developers can see problem areas, and what people like.[138]
However, the exact procedure of measuring this, is beyond the scope of this thesis.

Confidentiality:
The second quality attribute is about the confidentiality of the information. Even though (Forum 2008e) shows it in its framework, no definition is given yet.
It can be derived from JFC9, 10 and 11: all of them are involved in data security and somehow in data confidentiality.
Confidentiality itself has been seen in several sections in this thesis, one of the most important statements has been made by Metsaars in (Metsaars 2008b) about confidentiality, that confidential communications and data should be kept private. Stan has seen it in (Stan 2008a), that it is about preventing unauthorised disclosure of sensitive information. She has seen two types of confidentiality: traffic flow confidentiality and information confidentiality (see section 2.6.4 for more details).
Based on these findings we will define the quality attribute as follows:

> *"The confidential information assets should be protected at all times against unauthorised disclosure, furthermore the confidential network Communications should be kept un-transparent and protected against eavesdropping. Both should be accomplished by using inherently secure and open protocols and standards"*

Yet, how can one measure that all of confidential assets and traffic are actually still undisclosed? One cannot. The only thing that one can test is if the confidentiality is breached by checking if there is a successful threat in the form of an unauthorised access or network security breach (see sections 0, 2.3.3, 2.6.4, 2.6.6 and 2.6.9 for relevant background information). However, these tests are outside the scope of the thesis.

Integrity:
The third quality attribute is about the integrity of the information. Even though (Forum 2008e) shows it in its framework, no definition is given yet.
This quality attribute is related JFC 9, 10 and 11 just like the second. All of them are involved in data security and somehow in data integrity.
Data integrity is defined as a security service in section 2.6.4 as:

> *"Integrity assures that transferred messages are received as they are sent, with no duplication, insertion, modification, reordering, or replays. Also deletion or destruction*

[137] Source: http://www.springerlink.com/content/g744753360415047/fulltext.pdf?page=1, visited at 28-10-08.

[138] Source: http://en.wikipedia.org/wiki/Usability_testing , visited at 28-10-08.

of data is included in this service, so all the transferred data should arrive to the receiver. So, this service prevents the unauthorized alteration or destruction of transmitted data by unauthorized entities."

Seeing this definition, allows us to define our own quality attribute:

"The data is assured to be integer: no unauthorised duplication, insertion, modification, recording, deletion, destruction or replays will have been applied to all of the data in the collaborative information domain."

Again, this integrity cannot be tested by its own. All one can do is check for every information asset if the asset has encountered any of the unauthorised acts as stated in the definition. See for more information section 2.6.4.
The exact details of the measure processes are of the scope of this thesis.

Availability:
The fourth quality attribute is about the availability of the data. It has been defined in (Forum 2008e) as:

"A collaboration's information should not be rendered unavailable either by mistake or by an adversary. This implies that any 'at rest' encryption keys are escrowed, and that information is held in open-standard formats. "

(Forum 2008e)

This first part is logical: information should always be available. The second part means that the keys for the encryption should reside with third parties until they are necessary. The trust broker should be the party where they can reside. Last of the second part means that all of the information should be available in document formats that are available to the masses without any necessary commercial licensing.
The availability of the information can be monitored by intelligent agents that check if the information is still there. The rest is not so easy to check. All one can do is warn the trust broker if any user tries to save the information in specialised formats or if he/she holds keys to encryption by himself without notifying the trust broker. The exact details of these processes are out of the scope of this thesis.

Efficiency/Performance:
The fifth quality attribute is about the efficiency of the security in terms of impact on the performance of the systems. It has been defined in (Forum 2008e) as:

"Security measures do not greatly affect the latency, bandwidth, or total cost of data retrieval, storage, or transmission. This implies that collaborating partners must possess the means to rapidly access decryption keys for all data in their possession for which they continue to have access privileges, allowing rapid data retrievals and offline malware scans."

(Forum 2008e)

The first part of the quality attribute should give no problems at all. However, looking at the second part and combining it with the quality attribute of availability, one should immediately understand that the third party should be a fast responding, always available and reachable party, and that keys can be distributed to both the third party (the trust broker) as well as the organisation that possesses the information.
This quality attribute consists of two parts: the performance in terms of latency, bandwidth, total cost of data retrieval, storage and transmission and the concept of managing one's own keys, while delivering copies to the trust broker.
The first part can easily be tested by performance tests and measures that allow one to measure the performance of the systems, as already is being done in current organisations. The second part will need monitoring of the keys in order to measure if all of the possessed data

can actually be accessed at the required ease. One could also choose to measure the required time to retrieve and use the key at the targeted encrypted information asset.
The exact details of the measure processes are out of the scope of this thesis.

Effectiveness:
The sixth quality attribute is about the effectiveness of the security in terms of time, information domains and the safety provided to the information assets. In other words, how effective the measures of the COA framework allow one to protect and manage both the information assets and their protective measures. It has been defined in (Forum 2008e) as:

> *"The COA framework provides an effective approach to organizing and controlling secure data transport and storage among a wide range of existing and future corporate information systems."*
>
> (Forum 2008e)

Again, one cannot measure the effectiveness itself, yet one can measure the amount of ineffectiveness of the organisation and control of the secure data and transport. As soon as there is a breach in security and an information asset is compromised, then ineffectiveness has been shown. This means that the effectiveness can be expressed as the amount of ineffective and detected breaches and the amount of effective breaches and information asset compromises. However, the exact details of these measuring processes are out of the scope of this thesis.

Agility:
The seventh quality attribute is about the flexibility and agility that the COA framework should supply to those who adopt it. It has been defined in (Forum 2008e) as:

> *The COA framework takes into account the dimensions of timeliness and flexibility. It enables development of business-driven enterprise architectures that are appropriately flexible and adaptable to facilitate changes in business operations with optimal rapidity, and ease, with minimal disruption.*
>
> (Forum 2008e)

Last, but not least is the quality attribute aimed at agility. Yet, how can one measure the agility of an enterprise, or let alone the flexibility and timelessness? These are concepts that are very hard to measure. Many research projects have been done on this field such as (Tsourveloudis NC 2002; MacKinnon 2008). However, they are out of the scope of this thesis for now, due to the complexity.[139] The less complex metrics can be found in measuring the flexibility. As seen in (Sprott 2005), one can measure the flexibility by the following metrics in a SOA:

- **Number (and granularity) of services:** this can show the size of the management task, the duplication of functionality and the difficulties one will have with service discovery. The more duplication and difficulties, the lower the flexibility. One can use this metric when one is going to test the flexibility of the architectural planning tasks and the necessary governance activities.
- **Change impact:** by checking the impact of changes and the duration that a change costs, one can see the maintainability and the cost to customize or change the business. The longer the change takes and the lower the impact of that change is, the less flexible the whole architecture is. This metric can be used to test the flexibility of the architectural planning tasks and the change projects.
- **Relative service independence:** this will allow one to see the architectural quality and costs for reuse, assembly and maintainability: the higher the cost, the lower the flexibility.

[139] Even though they are of scope, the author does recommend reading them to further study the measureprocesses.

Furthermore, the more the (security) services depend on one another, the more inflexible the system will become.

All of the metrics will have to be measured before the implementation of the COA framework as a reference and again after the implementation to see whether the quality attribute will hold. However, the exact details of these measuring processes are out of the scope of this thesis.

3.8.3. Summary: the COA Quality Attributes

Looking back on this paragraph, we can conclude the following:

- **The COA framework quality attributes** can be used to measure if one has achieved the objective of the COA framework: to implement usable, manageable information security measures, which protect the confidentiality, integrity and availability of the information assets. The successful implementation of the COA framework will furthermore mean that the security measures will not have a negative impact on the information systems' performance and will still be effective. It will furthermore supply a flexible enterprise information architecture.
- **Usability/manageability:** the security measures should be easily understood and easily manageable. This can be tested by user test panels with specific procedures that are out of the scope for now.
- **Confidentiality:** confidential information should be protected at all times from unauthorised disclosure and the confidential network communications should be kept safe from eavesdropping and transparency. The only thing one can test is whenever there is a successful unauthorised access or whenever there is a breach in network security that could have allowed eavesdropping.
- **Integrity:** information assets should never be tampered with and always protected from attempts to do so. This can only be tested by accounting for every detected tampering with the information asset.
- **Availability:** the information assets should always remain available to the organisations that possess them. This can be measured by using monitoring intelligent agents and the trust broker.
- **Efficiency/Performance:** implementing the COA framework should not have any negative impact on the information systems performance. This can only be measured by measuring the performance before and after the implementation of the framework, allowing one to see the differences.
- **Effectiveness:** implementing the COA framework should create an effective approach to organizing and controlling secure data transport and storage among a wide range of existing and future corporate information systems. One can measure the ineffectiveness of the security and express the effectiveness as a percentage of 100 minus the ineffectiveness in terms of successful threats and attacks.
- **Agility:** business agility alongside with flexibility should be accomplished by implementing the COA framework. How one can measure agility is out of the scope of this thesis, yet the flexibility can be measured by the number (and granularity) of services, the change impact and the relative service independence.

3.9. COA technologies

3.9.1. Introduction

The last part of detailing the COA framework, is detailing the COA technologies. This will be done in the oncoming section (section 3.9.2) and then again summarised (section 3.9.3). However, one should notice that we cannot elaborate too much on the technologies due to both time constraints and the unavailability of completed studies surrounding these technologies and their fitness for the COA framework. So this paragraph should be seen as a small elaboration on the matter and nothing more. One should certainly do more research on the field of necessary technologies.
Most of the work here will be based on external sources and paragraph 2.6.

3.9.2. COA Technologies

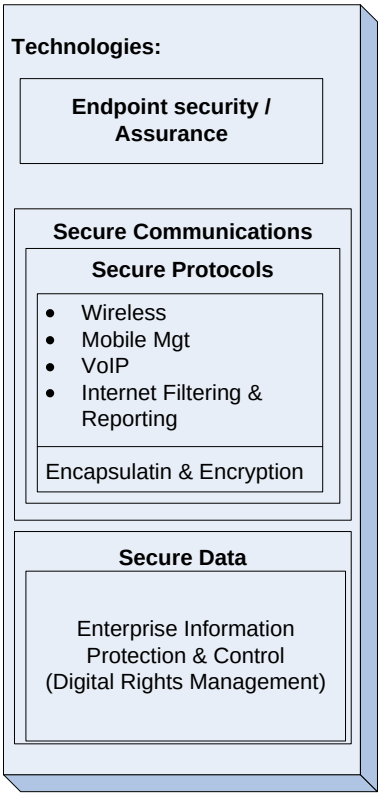


Figure 63: Technologies.

While conducting this study, we have found a set of technologies and even some brands and names that should be considered and further researched in the context of the COA framework. They have been grouped here, based on Figure 63:

- **End-point security management technologies:** these can vary from end-point security tools to complete management suites. The following technologies have been named by (Teheux 2008)[140]:
  - HP OpenView Select Access.
  - Oracle Access Manager.
  - IBM Tivoli Federated Identity Manager.
  - Sun Java System Access Manager.
  - The NACs from Cisco and Microsoft.
  - The Trusted Computing Groups Trusted Network Connect that is focusing on incorporating End-point Security within a Digital Rights Management (DRM).
  - Trusted Platform Module.

  The technologies on this field would become fully usable if they would be capable of realising all the related requirements and recommendations that have been given in sections 2.6.5, 2.6.4, 2.6.9, 3.6.5 and 3.7.3.
- **Secure communications technologies:** we have seen various technologies that could be used for securing the communications, such as:
  - *Tunnelling or securing protocols* such as IPSec, SSL, TLS and Kerberos. See section 2.6.4 for more details.
  - *WS-standards such as* WS-Secure Conversation. See section 0 for more details.
  - *Wireless security protocols* such as 801.2x, see intermezzo 6 in section 2.6.4 for more details.
  - *Internet filtering products* which should work in the cloud, ensuring that the context will be filtered at all times as also have been discussed in (Forum 2006d).
  - *VoIP security frameworks and technologies* that are in line with (Forum 2006a).
  - *Encryption methods and protocols* as have been explained in (Simons 2006; Stan 2008a).

[140] He noticed that all of them should be further researched and that none of them are fully suitable for the End-point Security service as should be implemented in the COA framework. See his work for more details.

- *Intelligent Application Gateways with firewalls and proxy servers* as have been discussed in section 2.6.4.

- **Secure data technologies:** we did not find any Jericho related studies that lined out exactly what would be necessary for secure data technologies. One should consider DRM-related technologies and other technologies, which are already available in a SOA such as XACML, XKMS, XrML and WS-Privacy.
- **Other technologies:** these technologies have not been shown in Figure 63. However, they should be considered for allowing the services and processes to do their work. The technologies enumerated here have been found in (Barannikov 2008; Bruning 2008a; Bruning 2008b; Teheux 2008):
  - *Identity Management:* the following technologies and frameworks have been found, for Federated Identity Management: SAML (2.0), WS-Federation and Liberty Alliance and for User centric Identity Management systems: OpenID, WS-* and the Microsoft Metasystem.[141]
  - *Reputation systems:* the following approaches have been found: the eBay feedback system, Jyte, Experian.
  - *Behaviour management*: the following approaches have been found: XDI and Link contracts and WS-Agreement.
  - *Authentication related systems*: the following approaches have been found: Microsoft Cardspace, Bandit, Higgins, OpenID and Extensible Resource Identifier.
  - *Other technologies, not related to these sources*: the WS*- protocols, SECRET and so on, and so forth. See the following works for more details (Demarteau 2008; Leijden 2008)

Technologies will have to be considered. Meaning that one should do a lot of further research on this field.
An interesting development is the fact that H. S. Teng and M. Plas are currently advocating a Jericho certification programme, in order to let the market come with the necessary technological solutions for de-perimeterisation. Hopefully, they will create a licensing programme that will consider the COA framework as well.
The COA framework Technologies can be used to actually implement or assist the services. End-point security tools can be used to implement the trust management service and the Device Lifecycle Management and Risk Management processes. DRM-related technologies can actually assist the Information Lifecycle Management processes and the Information Asset Management Services. The reader should be capable by now to see what other relations he can find between the technologies, services and processes of the COA framework.

3.9.3. Summary: COA Technologies

As one can see in the previous section: there are loads of technologies which will still need more research in order to check if they are usable for the COA framework and if any additional changes might have to be applied to make them usable.
There are four groups of technologies which need to be considered: endpoint security technologies, secure data technologies, secure communications and other technologies.

[141] We have seen other technologies as well, see section 2.6.7.

COA V2.0 UPDATES:

The following relevant additions have been made in the COA 2.0 revision:

- **The usage of VoIP**, in: "Position Paper VoIP in a de-perimeterised world", which can be summarized as follows:
  - *Need for open standards:* VoIP implementations should adhere to open standards without any royalties.
  - *The need for "Secure out of the box"-products:* all components in a VoIP implementation must be secure "out of the box" according to an industry agreed profile. The products should be capable of surviving the raw internet.
  - *The need for end user/mutual authentication:* since calls can be made from virtually anywhere, one will need support for strong mutual authentication.
  - *Capabilities of VoIP protocols:* the VoIP protocols should be capable of end-to-end encrytpion, business requirements (forwarding, conferencing, etc.), control and configuration of the device, updating/maintaining and remediating the device, end device and controller authentication, strong mutual user authentication.
- **More details on internet filtering and reporting,** in "Position Paper Internet Filtering & Reporting", which can be summarized as follows[1]:
  - *Necessary filtering capabilities:* a set of necessary filtering capabilities have been defined, such as URL filtering, wildcard capabilities, sufficiently granular categorization, intelligent handling and differentiation of port 80 tunneling traffic, blocking by computer name/individual user. It furthermore should screen all content to present it 100% malicious-content-free to the user.
  - *Requirements in service provision, loggin& reporting and systems management*: additional requirements have been described, see the paper for more details.
- **Additional information on the issues around wireless networks which have to be resolved**, in " COA Paper Secure Protocols – Mobile Management", which can be summarized as follows:
  - *No control over the network:* one is still unable to control the network experience (QoS, connection authentication, cost of the connection) of the user in a foreign (or public) networked environment.
  - *Need for transparency:* There is a need for the creation of a transparent connection, which could be realized by expressing the Wi-Fi hotspot "contract" electronically. The client must also transparently try to authenticate the network itself, such that an automatic decision can be made to allow a connection based on corporate or personal policies.
- **Introduction of the client to service VPN** in the new revision of "(The Need for) Inherently Secure Communications": As an addition to the statements above, the Jericho Forum advertises for a client to service VPN, in which applications use built in tunnel capability so that each protocol is isolated and only services/prots in use are exposed. See the paper for more details.

The publications can be found at: https://www.opengroup.org/jericho/publications.htm

3.10. Intermezzo: Mapping the COA framework to paragraph 2.6 and the commandments

This paragraph will provide a mapping between the COA framework and section 2.6 and the Jericho Forum Commandments in order to provide a little more transparency to the relation between the items of the COA framework and the previous work of the Jericho Forum. This mapping is not complete, it is just an approach to get a better overview of all of the Jericho related knowledge:

COA framework group	Name element	Relevant sections paragraph 2.6	Relevant Jericho Forum Commandments
Principles			
	Participating parties	2.6.6 (privacy), 2.6.7 (identification), 2.6.8 (trust), 2.6.9 (endpoints).	5, 6, 7 and 8.
	Trust	2.6.3 (core), 2.6.6 (classification) 2.6.8 (trust), 2.6.9 (endpoints).	5,6 and 7.
	Assurance	2.6.5 (Policies), 2.6.6 (privacy), 2.6.7 (identity/ reputation), 2.6.8 (trust broker), 2.6.10 (audit).	7.
	Risk	2.6.6 (classification).	3.
	Compliance	2.6.9 (endpoint security), 2.6.10 (audit).	1, 2, 3, 4, 5, 9, 10 and 11.
	Legal, regulatory, contractual	2.6.10 (audit)	3, 9, 10 and 11.
	Benefits and obligations	None, TBA.	None, TBA.
Processes			
	People Lifecycle Management	2.6.6 (classification), 2.6.7 (identities), 2.6.8 (trust, etc.).	1, 2, 3, 6, 7, 8 and 10.
	Risk Management	2.6.6 (classification).	1 and 3.
	Information Lifecycle Management	2.6.3 (core), 2.6.4 (message protection), 2.6.5 (information access policies), 2.6.6(classification, protection, privacy) 2.6.7 (authorisation), 2.6.8 (trust, broker), 2.6.10 (audit).	1, 2, 3, 9, 10 and 11.
	Device Lifecycle Management	2.6.5 (policies), 2.6.7 (identity), 2.6.8 (trust, broker), 2.6.9 (endpoint security), 2.6.10 (audit).	1, 2, 3, 4, 5 and 8.
	Enterprise relationship management	2.6.5 (policies), 2.6.6 (data classification, protection, privacy), 2.6.7 (identities), 2.6.8 (trust, broker), 2.6.10 (audit).	1, 2, 3, 6, 7 and 10.

Table 14: Mapping between COA framework elements, paragraph 2.6 and the Jericho forum commandments, part 1.

Services	Identity Management, federation and reputation	2.6.6 (privacy), 2.6.7 (Identity Management), 2.6.8 (trust, management, broker), 2.6.10 (audit).	1, 2, 3, 6, 7, 8 and 10.
	Trust management and classification	2.6.5 (policies), 2.6.6 (classification, privacy), 2.6.8 (trust, management, broker), 2.6.9 (endpoint security), 2.6.10 (audit).	1, 2, 3, 6, 7 and 10.
	Policy management	2.6.5 (policy management), 2.6.9 (security), 2.6.10 (audit).	1, 2, 3 and 5.
	Meta/ information management	2.6.5 (policy management), 2.6.6(information protection, management), 2.6.8 (trust, broker), 2.6.10 (audit).	1, 2, 3, 4, 5, 9, 10 and 11.
	Audit	2.6.8 (trust, broker), 2.6.10 (audit).	1, 2, 3, 6 and 7.
Attributes of the solution			
	Usability / manageability	Indirect: 2.6.2(core) , 2.6.4 (attribute named).	2
	Confidentiality	Indirect: 2.6.2(core) , 2.6.4 (attribute named).	10
	Integrity	Indirect: 2.6.2(core) , 2.6.4 (attribute named).	9, 10.
	Availability	Indirect: 2.6.2(core) , 2.6.4 (attribute named).	2
	Efficiency / performance	Indirect: 2.6.2(core) , 2.6.4 (attribute named).	1 and 2.
	Effectiveness	Indirect: 2.6.2(core) , 2.6.4 (attribute named).	None.
	Agility	Indirect: 2.6.2(core) , 2.6.4 (attribute named).	2
Technologies			
	Endpoint security	2.6.9 (Endpoint Security)	1, 2, 3, 4, 5, 6, 7 and 8.
	Secure communications	2.6.4 (protocols, standards)	1, 2, 3, 4, 5 and 6.
	Secure data	2.6.4 (measures, protocols), 2.6.6 (data classification, protection, management).	1, 2, 3, 4, 5, 9, 10 and 11.

Table 15: Mapping between COA framework elements, paragraph 2.6 and the Jericho forum commandments, part 2.

3.11. Application and adoption of the COA framework

3.11.1. Introduction

As we have tried to detail the COA framework in paragraph 3.5 till 3.9. We will now try to give a more detailed application of the framework (see Figure 64).
One should realise by reading those previous paragraphs that the elements of the COA framework are far from finished. Most of them will certainly need more research or even complete roadmaps to become workable solutions. This means that it will be impossible to give a complete description of how one should adopt the COA framework, since the elements of the framework are not completely clear yet and many of them will have to be researched some more.
Furthermore, there is a lack of resources and time to completely describe or select all of the necessary methodologies and scenarios for implementing them into the architectures. That is why some of the architectures (SaaS, Enterprise architecture) will be discussed in little more detail than others (Network architecture, system architecture).

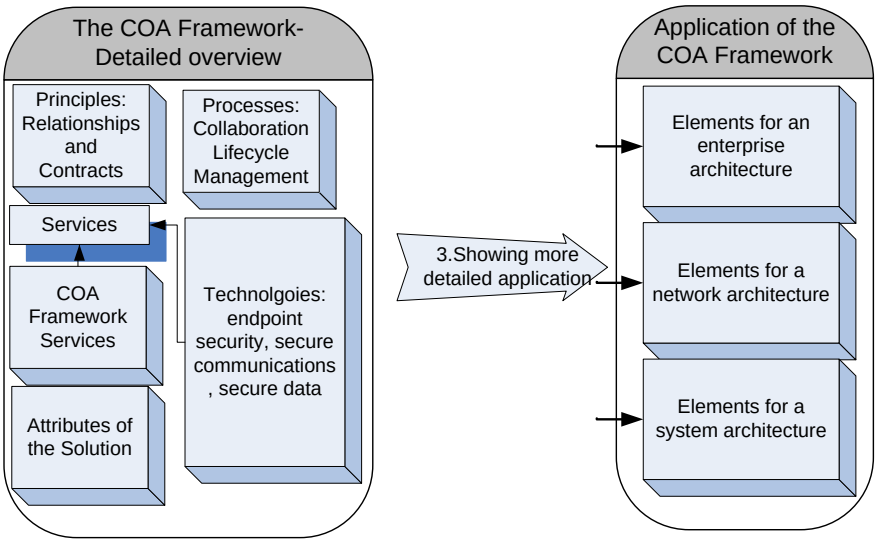


Figure 64: Overview of paragraph 3.11, detailing the COA framework application.

However, an incomplete framework does not refrain one from describing briefly how it could be adopted. Which is exactly what will be done in this paragraph:

- The adoption of the COA framework within SAAS will be described in section 3.11.2
- The adoption of the COA framework within an enterprise architecture will be described in section 3.11.3
- The necessary COA framework elements for a network architecture will be described in section 3.11.4
- The necessary COA framework elements for a system architecture will be described in section 3.11.5

We will summarise our findings in section 3.11.6.

3.11.2. Considering the COA framework and SaaS

Introduction:
This section has not been defined in terms of a research question. However, it does allow us to see some extra value of the COA framework as it could be the answer to the trust problems within the SaaS market.
So, in order to give a description of how one should implement the framework in a SaaS scenario, we will first have to understand which players can be identified in the SaaS environment and what their needs will be. As we have found ourselves quite an introduction to that environment in paragraph 2.3, we will now take it one step further, by using one of the SaaS scenarios defined in (Dirk Hanenberg 2008) and describe for each actor what kind of COA framework elements he will need to secure the collaboration.
However, we will not go into any details of the exact implementation of those COA framework elements, nor will we cover all of the scenarios that have been given in (Dirk Hanenberg 2008). So further research on this field should be done as well. This section will only give some practical insights in using and the value of the COA framework.
Before going to the scenario description, one should notice that we will try to create a situation in which the trustworthiness of the parties will be maximised, by using all the applicable COA framework elements. However, one could choose to go with less trust and implement less of the named COA framework elements.

The scenario – Integration as a Service:
The scenario that will be analysed is the "Integration as a Service" distribution network since it is seen as the most suitable implementation for both small and large Enterprises in (Dirk Hanenberg 2008). It works like this:
The model is based on an Integration Service Provider (ISP) that operates between the Independent Software Vendors (ISVs)[142] and the customers. See also figure Figure 65.
The ISVs can offer their services to the customers via the ISP and new ISVs can be added to the ISP by customer request, which needs to be approved by the ISP.
Each customer can use any service of an ISV if it is connected to the ISP. This allows for a "one to many" model in which one ISV can deliver is application to many customers via the ISP.

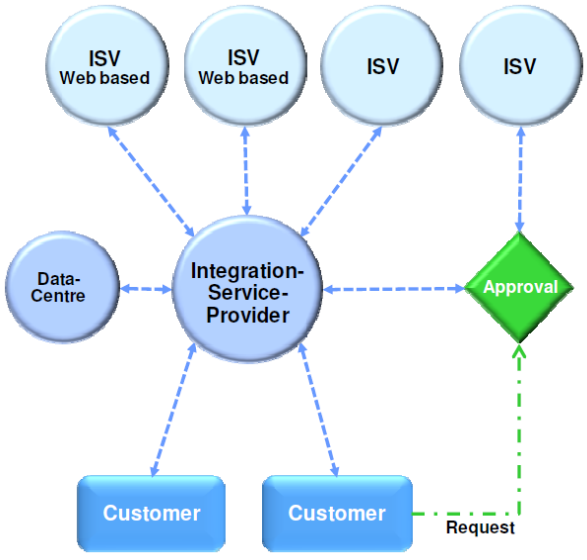


Figure 65: Distribution model 'Integration as a Service': distributive parties. (Dirk Hanenberg 2008)

[142] The idea is that there are web based ISVs that deliver their application directly to the ISP without human intervention and ISVs that do need human intervention.

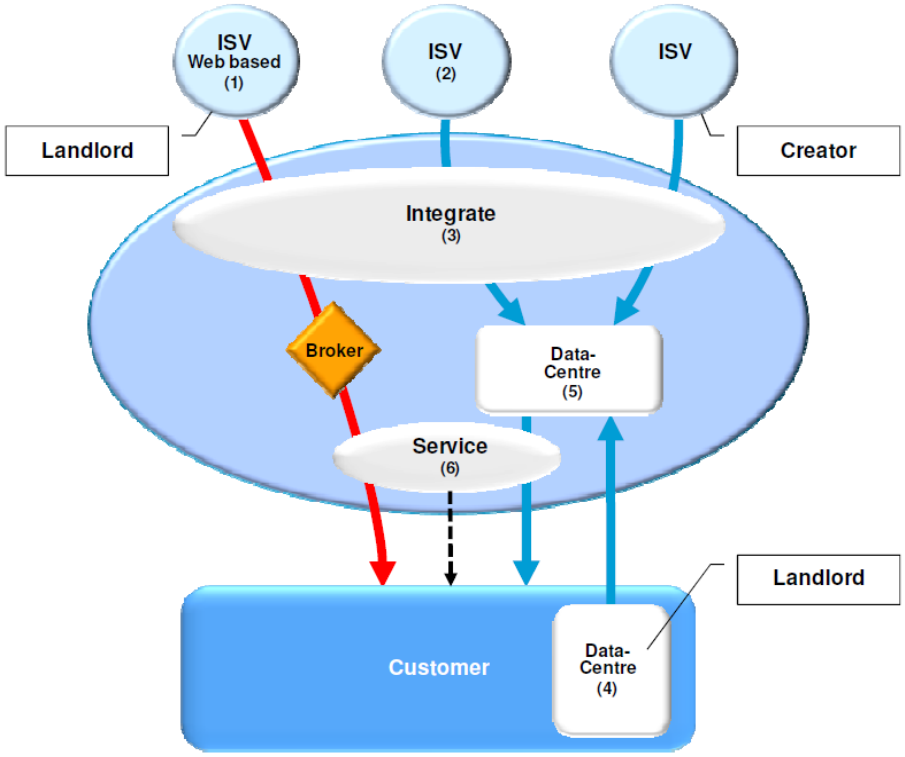


Figure 66: Distribution model 'Integration as a Service': flow of software services / data. (Dirk Hanenberg 2008)

In (Dirk Hanenberg 2008), we have found a more detailed description (see also Figure 66):

> *"When a customer requires a web-based application, the application will be provided directly from the ISV to the customer. In this situation, the ISP operates as a broker, as he matches the buyer and seller. The ISV of the web-based application is considered a landlord, as the ownership of the assets stays with the ISV. The right to use it is sold to either the customers or the ISP. When a traditional application is required it will be integrated with the other applications that are offered by the ISP to a specific customer. With a fully ASP-based customer organization as starting point of the feasibility study, it is likely that some applications of the customer are not offered by ISV. Therefore, it is also possible that applications are offered by the customer to the ISP. Because the ownership of those specific applications stays with the customer, the customer becomes a Landlord. The ISP stores all traditional applications in a data centre, which can be in-house or outsourced. As a result of this the ISP becomes responsible for the maintenance and security of assets. In addition to those services, the ISP is also responsible for the billing and contractual services towards the customer. The customer receives an invoice from the ISV for all acquired licenses per application. As a result of this the ISP is responsible for delivery of that application, and therefore the contact party in the Service License Agreement (SLA). Overall the business model of the ISP is considered a Distributor, as the ISP adds additional services to the products that are offered to the customer."*
>
> (Dirk Hanenberg 2008)

The players and their needs from the COA framework:

As we look at the scenario, we can identify the following parties and their needs:

- **The broker**: The broker in this situation is the ISP. He is the facilitator between both the providers and the consumers.

- Description: “A broker facilitates sales by matching potential buyers and sellers. Unlike a typical distributor, a broker does not take ownership of the product. Instead the broker receives a fee from the buyer, the seller or both.“ (Dirk Hanenberg 2008)
- Needs from a de-perimeterised point of view:
  - To ensure that the applications of the ISV are safely integrated.
  - To ensure that the ISVs are trustworthy players.
  - To ensure that the customers will be trustworthy payers.
  - To ensure that the ISVs will not spread the customers data.
  - To ensure that the customers will offer their own applications safely and secure.
  - Ensure the security of the information assets of the customers.
  - To ensure a safe collaboration with both customers and ISVs, based on a two way trust.
  - To ensure a secure and workable delivery of the application.
  - To ensure that the PII of the customers and the ISVs is handled properly.
  - To ensure that the applications will work in the unsafe environment of the internet.
  - To ensure that their organisation will be trustworthy in terms of personnel and devices.
  - To ensure that their processes are in line with legal and regulatory law.
- *Necessary COA framework elements*: one could argue that the following Framework elements will be necessary to fulfil the needs:
  - The principles: all parties need to know each other, trust each other and gain assurance of each other that all agreements will be followed, as well as legal and regulatory law, privacy needs and security compliancy.
  - The services:
    - Identity Management Federation and Reputation to take care of the identities of their own personnel and hardware.
    - Trust Management and Classification to handle the external identities of both customers and ISVs and to classify all of the data assets that are created by their own organisation. Later on, the trust management service could also be used for handling the connection with the PII broker and the trust management framework. This will allow the broker to gain transparency in the trustworthiness of the ISVs and customers. It will also allow to check how the privacy is handled via the PII broker.
    - Policy Management for handling the information access policies of all of the information assets and to provide information about the policies. This will allow the ISP to see whether the ISVs and customers have applied the correct policies to the assets.
    - Meta/Information Management for handling all the information assets that will be used or provided by or exchanged with[143] their own organisation.
    - Audit to audit both ISVs and important (application serving) customers in order to check whether their processes and results are in line with compliancy, contractual, legal and regulatory law. This will provide the capability to check for the assurance which is necessary for the other needs of the broker.
  - The processes:
    - The People Lifecycle Management processes for handling their own personnel and externals such as the customers and the ISVs.
    - The Risk Management for managing and assessing the risk as being the broker.
    - The Information Lifecycle Management for handling their own information assets as well as those that they provide. This could also include auditable data of events (such as transactions) and data from customers and ISVs.
    - The Device Lifecycle Management to handle their own endpoints and later on all of their devices.
    - The Enterprise Relationship Management to handle all of the relationships with (optional other ISPs,) the ISVs and the customers.

[143] This exchanging can vary from inter-organisational exchanges to intra-organisational exchanges.

- Technologies such as Internet Filtering and Reporting for cleaning the traffic, for the customers, Encapsulation and Encryption for safeguarding the connection, DRM mechanisms for safeguarding the data.[144]
- Attributes of the Solution: all of them could be required. They reflect a good implementation of a security system.

- **Landlord**: The landlord in this situation is the ISV. He has the ownership of the application and rents it to customers. It could happen that the customer becomes a landlord as well by providing his own necessary applications to the ISP.
  - Description: "A Landlord sells the right to use, but not to own, an asset for a specified period of time. The term 'landlord' is used in a general sense because it does not apply merely to physical assets but also to virtual assets and services as well." (Dirk Hanenberg 2008)
  - Needs from a de-perimeterised point of view:
    - To ensure a safe and trustworthy relationship with the ISP and guarantees in terms of payment (mutual trust).
    - To ensure that the application will survive the hostile world of the internet.
    - To ensure that the applications that they provide are safe and secure (by) themselves.
    - To ensure the PII of the customers, themselves and the ASPs is handled properly.
    - To ensure that their processes are in line with legal and regulatory law.
    - To ensure the safety of the data that is processed by their hardware and applications.
    - In cases of customers providing (shared)applications: that they cannot use the data of other customers for their own benefit without the consent of the other customers.
  - *Necessary COA framework elements*: One could argue that the following Framework elements will be necessary to fulfil the needs:
    - The principles: All parties need to know each other[145], trust each other and gain assurance of each other that all agreements will be followed, as well as legal and regulatory law, privacy needs and security compliancy.
    - The services:
      - Identity Management Federation and Reputation to take care of the identities of their own personnel and hardware.
      - Trust Management and Classification to handle the external identities of both customers and ISP(s) and to classify all of the data assets that are created by their own organisation. Later on, the trust management service could also be used for handling the connection with the PII broker and the trust management framework. This will allow the broker to gain transparency in the trustworthiness of the customers and ISP. It will also allow to check how the privacy is handled via the PII broker.
      - Policy Management for handling the information access policies of all of the information assets and to provide information about the policies. This will allow the landlord to see whether the ISP is modifying the policies or not.
      - Meta/Information Management for handling all the information assets that will be used or provided by or exchanged with[146] their own organisation.
      - Audit to audit the ISP in order to check whether their processes and results are in line with compliancy, contractual, legal and regulatory law. This will provide the capability to check for the assurance which is necessary for the other needs of the landlord.
    - The processes:

[144] One could argue that they will need VOIP, wireless security, et cetera for their own infrastructure. However, that will remain of scope for now.

[145] One could argue in this scenario that, if the ISP will guarantee the payment, that the ISV does not necessarily needs to know the customers.

[146] This exchanging can vary from inter-organisational exchanges to intra-organisational exchanges.

        - The People Lifecycle Management processes for handling their own personnel and externals such as the (optional: the customers) and the ISPs.
        - The Risk Management for managing and assessing the risk as being the provider of an application.
        - The Information Lifecycle Management for handling their own information assets as well as those that they provide. This could be the information that is stored by an application in terms of event logging or data from the customers.
        - The Device Lifecycle Management to handle their own endpoints and later on all of their devices.
        - The Enterprise Relationship Management to handle all of the relationships with the ISPs, customers and other ISVs.
      - Technologies such as encapsulation and encryption for providing secure communications and DRM tools for providing safety of the information assets.
      - Attributes of the Solution: All of them could be required. They reflect a good implementation of a security system.
- **Customers:** The customers where not discussed in (Dirk Hanenberg 2008), however, they should not be forgotten in the collaborative environment. That is why they will be discussed here as well.
  - *Description:* These are the "non-landlord" customers which do not supply any applications themselves. They simply use the applications that have been provided by the ISP and might provide data by their own data warehouse to provide all of the information online.
  - Needs from a de-perimeterised point of view:
    - To ensure that they can use the applications that they rent, whenever they want to.
    - To ensure that their PII and other data is handled with care, according to the classification of the data.
    - To ensure that their data will not get lost or modified or accessed by anyone other then themselves.
    - To ensure that the ASP(s) and ASVs are trustworthy and that they have a mutual trust relationship (between the ASP(s) and ASVs as well as between the customer and the ASP(s) and between the ASV and the customer).
    - That their identities is handles with care.
    - That all of these needs are ensured, irrespective of the location where their data, applications, processes and identities are handled.
  - *Necessary COA framework elements:* The necessary COA framework elements for a customer will highly vary: if it is just an individual consumer, using one simple SaaS service for which he only needs an identity and a manual classification approach, then one could argue that he will only need the principles and the Identity Management service. However, if it is a customer which consists of an organisation with multiple identities using multiple SaaS services, then one could define his needs as follows:
    - The principles: All parties need to know each other, trust each other and gain assurance of each other that all agreements will be followed, as well as legal and regulatory law, privacy needs and security compliancy.
    - The services:
      - Identity Management Federation and Reputation to take care of the identities of their own personnel and hardware.
      - Trust Management and Classification to handle the external identities from the ISP(s) and to classify all of the data assets that are created by their own organisation. Later on, the trust management service could also be used for handling the connection with the PII broker and the trust management framework. This allows the customer to check whether the parties he is working with are trustworthy or not and whether his PII is handled correctly.
      - Policy Management for handling the information access policies of all of the information assets and to provide information about the policies.

- Meta/Information Management for handling all the information assets that will be used or provided by or exchanged with[147] their own organisation.
- Audit to audit the ISP in order to check whether their processes and results are in line with compliancy, contractual, legal and regulatory law.

▪ The processes:
- The People Lifecycle Management processes for handling their own personnel (and optional: the personnel of the ISP).
- The Risk Management for managing and assessing the risk that comes from using the applications and processing their own information assets.
- The Information Lifecycle Management for handling their information assets.
- The Device Lifecycle Management to handle their own endpoints with which they access the applications on the net via the ISP and later on all of their devices.
- The Enterprise Relationship Management to handle all of the relationships with the ISP(s) (,optional: other customers) and the ISVs.

▪ Technologies such as encapsulation and encryption for providing secure communications and DRM tools for providing safety of the information assets.
▪ Attributes of the Solution: all of them could be required. They reflect a good implementation of a security system.

Looking back and forward:
All of the COA framework elements could be of rather good use here. If all parties decide to use them as described here, then it would allow the parties to have enough means that provide a solid base of trusting one another. The identities, trust states of devices, reputations, accountability data et cetera, all would help to provide means and assurance for a trustworthy relationship. The ISVs, ISP and customers will be capable of controlling each other by many means such as executing audits, manage each other their information assets and by providing enough information about (information access) policies. The trust broker itself could later on provide enough additional means of assurance in order to ensure a trustworthy relationship.

However, they will not be capable of managing the PII and the reputational information on the short term, due to the lack of a trust management framework and a PII broker. Until then, one can use their trust management services in a P2P fashion to be capable of the required management actions such as: handling one's identity, checking one's device, exchanging accountability information et cetera.

However, it will not be easy to implement the COA framework, because of the missing details and missing components. That is why the following recommendations have been defined:

- **Develop a consumer version of the COA framework:** As the COA framework is aimed at helping organisations, it is not focussed on assisting the individual consumer. It would be wise to assist all the actors in the environment and thus develop a consumer version of the COA framework. Allowing the consumer to make use of the business services, without having to go through many complexities, in an environment where all the other parties of the transaction chain have already adhered to the COA framework.
- **Use the early recommended roadmaps to create a new one for the SaaS broker, landlord and consumers:** Many of the elements of the COA framework elements will require further research and development as one has seen in the earlier paragraphs (3.6 -3.9). Most of them should be developed according to a roadmap, allowing for a more sophisticated solution over time. These roadmaps can be used to develop a set of roadmaps for the players in this SaaS community, so that each player can evolve to a COA, allowing safe collaboration while resolving the SaaS trust issues.

[147] This exchanging can vary from inter-organisational exchanges to intra-organisational exchanges.

- **The creation of open and inherently secure standards:** All of the products from the COA framework should be developed based on open inherently secure standards and protocols, allowing safe exchange of information (assets) and services between the parties in the SaaS community. This also counts for the implementation processes that will have to be developed to create a usable implementation methodology allowing the SaaS players to implement the COA framework.
- **Further research:** There is still a lot more research to be done on this field. One will have to detail this scenario and see what other kind of implementations could be viable (allowing one for still having enough mutual trust to guarantee a safe collaborative success of the SaaS market). Furthermore, one should research the other scenarios as defined in (Dirk Hanenberg 2008).

3.11.3. Implementing COA framework elements in an enterprise architecture

Introduction:
This section will partially answer research question number three ("How can an architecture for a system, for a network and for an enterprise adopt the COA framework?"). First, we will discuss what an enterprise architecture is. Second, we will discuss what COA framework elements will be required and how they could be implemented. We will end the section by giving a set of recommendations about which steps will have to be taken next in order to be capable of implementing the COA framework into the information architecture that is discussed here.

What is an enterprise architecture?
The definition of an enterprise architecture is three-fold: if one wants to know what an enterprise architecture is, one will first have to understand what an enterprise is, which will define the scope of the architecture. Next, one will have to know what the architecture is, which describes the artefact itself. Finally, combining them, allows one to see what the enterprise architecture is.
An Enterprise can be defined as:

> *"any collection of organizations that has a common set of goals and/or a single bottom line. In that sense, an enterprise can be a government agency, a whole corporation, a division of a corporation, a single department, or a chain of geographically distant organizations linked together by common ownership. The term "enterprise" in the context of "enterprise architecture" can be used to denote both an entire enterprise, encompassing all of its information systems, and a specific domain within the enterprise. In both cases, the architecture crosses multiple systems, and multiple functional groups within the enterprise...." "...Large corporations and government agencies may comprise multiple enterprises..."*
>
> (OpenGroup 2005)

In other words, the scope of the enterprise architecture can vary from a part to a whole of the organisation, yet it will always cross multiple systems and functional groups.
The architecture can be defined as follows:

> *"The fundamental organization of a system, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution...""... in TOGAF, "architecture" has two meanings depending upon its contextual usage: A formal description of a system, or a detailed plan of the system at component level to guide its implementation. The structure of components, their inter-relationships, and the principles and guidelines governing their design and evolution over time."*
>
> (OpenGroup 2005)

Knowing these definitions allows us to define our enterprise architecture:

> *"A formal description of, or a detailed plan of a (part of a) corporation, with its information systems, domain(s) and multiple functional groups"*

In other words: it describes a part of the enterprise IT, its business processes, its functional groups et cetera.

Which elements of the COA framework will be required?
Even though we know what an enterprise architecture could comprise, we still do not know what elements of the COA framework would be necessary. It should be clear that the necessary components will depend on the scope of the architecture. If the architecture would comprise a complete corporation, then it will definitely be the case that all of the COA framework elements will be a necessity. However, if one would only comprise a part of the corporation, say the cleaning department, then that would not be the case.
Yet, there is more: depending on the approach of defining the architecture or how one wants to do business, one will have to comprise different aspects of an enterprise architecture, making it even harder to define the required elements of the COA framework.

Seeing the difficulties ahead, we should take an approach which allows us to comprise most of the architectural elements and visions. In other words, we should try to take a holistic approach or methodology in defining an enterprise architecture and describing all of its materials. TOGAF gives such a methodology.
TOGAF uses the TOGAF Architecture Development Methodology (ADM) to describe the complete enterprise architecture. We will look briefly at it and describe which of the COA framework related actions should be carried out during the phases of the ADM:

- Preliminary Phase:
  - *Description:* Framework and Principles: In this phase one can prepare the organization for successful TOGAF architecture projects.
  - COA related actions and elements:
    - one should discuss / investigate the COA framework in this phase as this would be a good preparation for changing the focus of the architecture by using the COA framework principles.
- Requirements Management:
  - *Description:* Every stage of TOGAF project should be based on and validate business requirements.
  - COA related actions and elements:
    - Think about the COA framework and its implications for governance and other concepts which are important in this phase.
    - Implement the COA framework attributes named in phase A, if they are already clear before one will start at phase A.

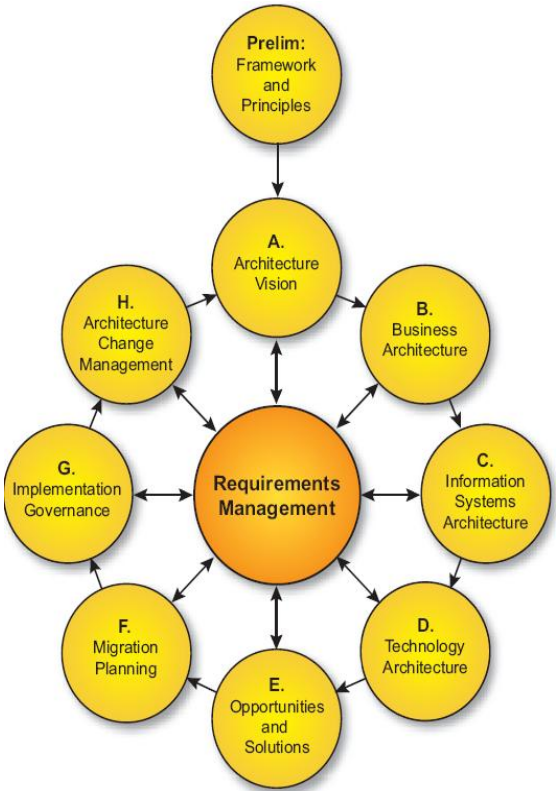


Figure 67: the ADM cycle. (OpenGroup 2005)

- Phase A:
  - *Description:* Architecture Vision: Set the scope, constraints, and expectations for a TOGAF project. Validate the business context and create the Statement of Architecture Work.
  - COA related actions and elements:
    - Use the COA framework Principles as a part of the strategic goals: If one wants to be capable of using the COA framework Principles throughout the strategy and its implementation in the enterprise, then one will have to use the principles at this point and transform them into business requirements.
    - Choose a set of COA framework Quality attributes and translate them to usable objects for this phase: The quality attributes are used to check whether one has achieved his goal in terms of typical quality attributes or the security. However, if one wants to achieve these goals, then they should be clear from the beginning, preferable as business requirements.
    - Choose the basic needs of the COA framework: In order to become a COA, one will have to implement the COA framework elements. This is the phase to select the necessary processes, services and technologies that will be necessary for the Business architecture and the technology architecture that needs to be described in a preliminary version.
    - Jericho concepts: Take the aspects of (i.e. Jericho Forum Commandments) and drivers for de-perimeterisation and see if they can be linked to the current business drivers, to see how important the COA framework will be to the enterprise and its architecture.
- Phase B:
  - *Description:* Business Architecture: Develop a Business architecture with a Baseline ("as is") and Target ("to be").
  - COA related actions and elements:
    - Define an implementation of the COA framework Processes: Work through each process and define the organisational structure which will be necessary for the process and additional the business goals and objectives, business functions, services, processes and roles which are related to the COA framework Processes. This automatically means that one will need to use roadmaps and see which standards of the processes can be used.
    - Define an implementation of the COA framework Services: Work through each service and define the organisational structure which will be necessary for the process and additional the business goals and objectives, business functions, services, processes and roles which are related to the COA framework Services. This automatically means that one will need to use roadmaps and see which standards of the services can be used.
    - Integrate the services and processes: Integrate the COA framework Services and – Processes in such a way that they are becoming reusable building blocks which can be mapped to the organisational units. Try to estimate how many times they will be used and what information the building blocks will need. Again, one should adhere to the open standards in these estimations.
    - Define COA building blocks: Identify those services which are based on the COA framework to be identified as COA building blocks, to show that the architecture entails to the COA framework. Apply also the details to them and see which standards will be necessary and if they can be based on old services from the current enterprise architecture which is in use.
    - Review: Review the (non) COA framework elements with stakeholders in the collaborative environment and check if they are in line with the standards and the methodology used by the collaborative environment.
    - Use COA quality attributes: Use the COA quality attributes as quality attributes to specify service levels.
- Phase C:
  - *Description:* Information Systems Architectures: Develop an Information architecture with a Baseline ("as is") and Target ("to be").
  - COA related actions and elements:

- Use COA framework elements for data definitions: Use the created standards for the COA framework Services and – Processes to define what data will be necessary to support the COA framework supporting elements. Use that as an input beside the other models that are used.
- Use Information Lifecycle Management processes: One should take the Information Lifecycle Management process as the primary data management process model and as a set of building blocks to ensure that the data management processes have been covered in line with the COA framework.
- Use Information Asset management Service: One should use the Information Asset Management service as one of the primary services and set of building blocks for handling data to ensure that all data will be handled properly according to the COA framework.
- Ensure that selected Data Architecture Resources and Vision are in line with the principles of the COA framework and the prospects around de-perimeterisation.
- Ensure that all stakeholders in the collaborative environment have been heard: In order to ensure a maximised value of the COA framework and a maximum compatibility of the organisation and their environment in terms of safe/secure information exchange.
- Define COA building blocks: Identify those services which are based on the COA framework to be identified as COA building blocks, to show that the architecture entails to the COA framework. Apply also the details to them and see which standards will be necessary and if they can be based on old services from the current enterprise architecture which is in use.
- Review: Review the (non) COA framework elements with stakeholders in the collaborative environment and check if they are in line with the standards and the methodology used by the collaborative environment.
- Use COA quality attributes: Use the COA quality attributes as quality attributes to specify service levels.
- Use COA Technologies: Use the applications of the COA Technologies to define which applications will be necessary, which are already in use and which need to be implemented and describe the further details as defined in (OpenGroup 2005).
- Define to be used applications for COA Services: Use the roadmaps and developed standards for selecting the applications which will be necessary to execute the COA framework Services. One should take in mind that opinions and standards can differ on several fields such as the necessary applications for data classification. So they should be selected based on the enterprise its strategy and vision.
- Define to be used applications for COA Processes: Use the roadmaps and developed standards for selecting the applications which will be necessary to execute the COA framework Processes.

- Phase D:
  - *Description:* Technology Architecture: Develop a Technology architecture with a Baseline ("as is") and Target ("to be").
  - COA related actions and elements:
    - Use COA Technologies: One should further refine the to be used COA framework Technologies and apply them to the technology architecture as described in (OpenGroup 2005).
    - Refine to be used applications for COA Services: One should further refine the to be used COA technologies and applications to be capable of executing the COA framework Services and apply them to the technology architecture as described in (OpenGroup 2005).
    - Refine to be used applications for COA Processes: One should further refine the to be used COA technologies and applications to be capable of executing the COA framework Processes and apply them to the technology architecture as described in (OpenGroup 2005).
- Phase E:

- *Description:* Opportunities and Solutions: Identify major implementation projects.
- COA related actions and elements:
  - Check for existing COA framework elements: Check which elements of the COA framework already have been implemented according to the current valid standards based on the roadmaps.
  - See which still need to be build: Check for which COA framework elements still have to be build and implemented into the enterprise.

- Phase F:
  - *Description:* Migration Planning: Analyze cost benefits and risk. Produce implementation roadmap.
  - COA related actions and elements:
    - Use roadmaps of the COA framework and define one: Check, based on the roadmaps of the different elements, which COA framework elements still need to be implemented in the enterprise. Create a roadmap of projects which still need to be executed in order to implement all of the necessary COA framework elements such as the processes, services and technologies. Prioritise the projects based on the other prioritisation done in this phase. Ensure that the roadmap is build according to the principles defined in (OpenGroup 2005).
    - Review: Review the roadmap with stakeholders in the collaborative environment and check if they are in line with their current roadmap and priorities.
- Phase G:
  - *Description:* Implementation Governance: Architecture Contracts are prepared and issued by the Implementation Governance. Board to ensure that the implementation project conforms to the architecture.
  - COA related actions and elements:
    - Detail the projects: Detail the COA framework related projects and show that they are COA framework related in order to allow one to see whether the enterprise architecture becomes a COA or not.
    - Check the impact: Check for the impact of the COA framework related projects, both on the organisation as well as on the collaborative environment.
    - Use principles and quality attributes: Use the COA framework Principles and – Quality attributes again to define acceptance criteria.
    - Optional: Use if available, the COA framework Risk Management service to asses the risks of the project and the upcoming issues.
- Phase H:
  - *Description:* Architecture Change Management: Ensure that the architecture responds to the needs of the enterprise.
  - COA related actions and elements:
    - General implementation: Ensure that there is a set of change management processes that is compatible with the COA framework and the new scope due to de-perimeterisation.

(OpenGroup 2005)

Next steps – Recommendations:

We have seen what an enterprise architecture is and what roughly needs to be done in order to implement the COA framework while using the TOGAF ADM. However, more research will have to be done in order to be capable of actually implementing the COA framework. Therefore, the following recommendations have been defined:

- **Create detailed standards:** The standards for the processes and services will have to be detailed, in order to ensure a correct implementation in the enterprise architecture. The standards considering the services should not just be detailed in terms of processes or service-building blocks, they should be detailed in terms of data and data flows as well. All of

the standards of the recommended roadmaps should be detailed as max as possible, ensuring interoperability between the solutions adhering to these standards.

- **Develop new reference data architectures:** A new reference data architecture set will have to be developed which takes the vast amounts and streams of data in mind which will come with de-perimeterisation. This will be necessary in order to support phase C of the ADM.
- **Further research and detail the implementation processes:** The briefly discussed steps will have to be detailed in order to be capable of implementing the services and processes.
- **Create additional standards to TOGAF for implementation:** The current TOGAF standard will have to be expanded with an additional COA standard in order to standardise the implementation processes of the COA framework elements.
- **Use roadmaps:** See phase F.

3.11.4. Implementing COA framework elements in a network architecture

Introduction:
This section will partially answer research question number three ("How can an architecture for a system, for a network and for an enterprise adopt the COA framework?"). First, we will discuss what a network architecture is. Second, we will take a look at what kind of COA framework elements will be required. Third, we will end the section by giving a set of recommendations about which steps will have to be taken next in order to be capable of implementing the COA framework into the information architecture that is discussed here.

What is a network architecture?
The Network architecture has been defined in (Glenn Hanson 1994) as:

> *"1. The design principles, physical configuration, functional organization, operational procedures, and data formats used as the bases for the design, construction, modification, and operation of a communications network...."" 2. The structure of an existing communications network, including the physical configuration, facilities, operational structure, operational procedures, and the data formats in use."*
>
> (Glenn Hanson 1994)

In other words, the network architecture describes the communication network and related concepts such as physical configurations, facilities, organisational units and necessary data.

Which elements of the COA framework will be required?
A COA network would be de-perimeterised. Which already gives us some insight in what elements should be necessary for a network architecture[148] such as:

- **The COA framework Principles and -Attributes:** The Principles and Attributes of the Solution should be used as principles for the network architecture. This allows one to create a de-perimeterised network which will be ready for the age of de-perimeterisation, provide QoS borders and proper means for collaboration.
- **The COA framework Processes:** As most of the processes are service based and not network based, they will not be implemented in the network architecture:
  - *Risk Management:* one will still need to assess the risks and threats at its network and take risk reducing actions for as far as possible.
- **The COA framework Technologies:** As many of the COA framework technologies are actually part of the network infrastructure they can be found in here:
  - *Wireless / Mobile Mgt :* As wireless has become a commodity and the usage of wireless mobile devices is increasing over time, one will need to manage these in a de-

[148] The insights can be found in paragraph 2.6.

perimeterised fashion. That is why they will have to be implemented in the Network Architecture.
- *VoIP:* As many POTS systems are replaced by VOIP systems which are used intensively, exchanging many data assets, one should manage these in a de-perimerised fashion as well, making the COA framework security technologies indispensable for a network architecture.
- *Internet Filtering and Reporting:* In order to be capable of handling the vast amount of threats that can come from the internet, one should implement these technologies in the network architecture as well.
- *Encapsulation and Encryption:* In order to be capable of having secure end to end communications, one should implement the technologies for allowing so in the network architecture.
- *Endpoint security measures:* These technologies will definitely be necessary in the network architecture as some devices of the network architecture can become endpoints.

The COA framework Services are not network based, the network only needs to be capable of supporting the communications of these services, allowing them to communicate freely. This also counts for the other processes besides the Risk Management processes and the COA framework secure data Technologies. That is why both of these will not have to be implemented in the network architecture.

Next steps – Recommendations:
We have seen what a network architecture is and what roughly needs to be implemented in order to make it a COA. However, more research will have to be done in order to be capable of actually implementing it. Therefore, the following recommendations have been defined:

- **Define the implementation process:** One will have to create a implementation process in order to implement the COA framework elements, allowing a network architecture to become a COA.
- **Use roadmaps:** As some of the necessary technologies will still need further research and development, it would be wise to use roadmaps which allow for a systematic evolution of the network architecture and its components.
- **Create a series of open and inherently secure standards**: In order to further accelerate and ease the adoption process of a COA network among organisations, one should develop a series of open standards that describe the COA framework elements and their implementation in a network architecture.
- **Further research:** To allow for all of the recommendations made above, one should further research the network architectures, their necessary COA framework elements and the required implementation processes.

3.11.5. Implementing COA framework elements in a system architecture

Introduction:
This section will partially answer research question number three (“How can an architecture for a system, for a network and for an enterprise adopt the COA framework?”). First, we will discuss what a system architecture is. Second, we will take a look at what kind of COA framework elements will be required. Third, will end the section by giving a set of recommendations about which steps will have to be taken next in order to be capable of implementing the COA framework into the information architecture that is discussed here.
What is a system architecture?
The problem with a “system architecture”, is that everything can be reviewed as a “system” from a abstract point of view. That is why we will define our own type of system for which the COA framework will be implemented:

> *"a system is a (set of) IT component(s) which will execute a set of processes either embedded or not. Either distributed or centralised in a single unit."* [149]

This allows us to see what a system architecture could be. We can create a definition of the system definition and the architecture definition of section 3.11.3:

> *"A formal description of, or a detailed plan of (part of a) (set of) IT component(s) which will execute a set of processes embedded or visible, either distributed or centralised in a single unit."*

This "detailed plan" in this context will also comprise the goals of the system, the environment where it is working in and the necessary technical capabilities.[150]

Which elements of the COA framework will be required?
The given definition of a system architecture is still allowing for multiple interpretations. There are many types of systems, many sets of IT components with many different goals and/or purposes. Therefore it is important to realise that any element of the COA framework could be a necessity, based on the type of system. Knowing this and understanding that we have a limited amount of time and resources, we still tried to define the necessary elements in a broader perspective, allowing the reader to reason if it would be a necessity in his system architecture as well:

- **The COA framework Principles and -Attributes:** The Principles and Attributes of the Solution should be used as design requirements for the system and its security. This will allow one to ensure that the architecture its implementation will be capable of surviving in a de-perimeterised environment, while delivering a maximum value in terms of safety, collaborative capabilities et cetera.
- **The COA framework Processes:** The necessity of the COA framework Processes will differ per type of system and its goal. However, we found the following processes directly linked to a system architecture:
  - *People Lifecycle Management support*: The system will have to support the data that comes from the PLM for as far as it is involved in the process. Therefore, one should check how these processes are involved and how support for them should be implemented. If one of the systems comprises the execution of a part of the processes, then that system should be classified as a COA supporting system and the necessary COA building blocks for the respective processes should be implemented in the architecture.
  - *Risk Management support*: The system will have to support the Risk Management processes (i.e. assessment and management processes) in order to allow the Risk Management Processes to comprise all of the elements of the collaborative environment.
  - *Device Lifecycle Management support*: The system will have to support all of the elements of the Device Lifecycle Management in order to be a part of that lifecycle. Therefore, one should check how these processes are involved and how support for them should be implemented. If one of the systems comprises the execution of a part of the processes, then that system should be classified as a COA supporting system and the necessary COA building blocks for the respective processes should be implemented in the architecture.
  - *Information Lifecycle Management support*: As the system will have to work with information assets, it should support all of the ILM elements. Therefore, one should check how these processes are involved and how support for them should be implemented. If one of the systems comprises the execution of a part of the processes, then that system

[149] This definition is loosely based on: www.sei.cmu.edu/opensystems/glossary.html, www.gemi.org/hsewebdepot/Glossary.aspx, nces.ed.gov/pubs98/tech/glossary.asp and www.exio.com/en/US/docs/interoperability_systems/c_ipics/101/pmc/user/guide/ippmcgl.html. All visited at 18-11-2008.

[150] Source: http://www.scribd.com/doc/2206800/SABSA-White-Paper , visited at 18-11-2008.

should be classified as a COA supporting system and the necessary COA building blocks for the respective processes should be implemented in the architecture.
    - *Enterprise Relationship Management support:* The system will have to support the data that comes from the ERM for as far as it is involved in the process. Therefore, one should check how these processes are involved and how support for them should be implemented. If one of the systems comprises the execution of a part of the processes, then that system should be classified as a COA supporting system and the necessary COA building blocks for the respective processes should be implemented in the architecture.
- **The COA framework services:** The necessity of the COA framework Services will differ per type of system and its goal. However, we found the following services directly linked to a system architecture:
    - *Identity Management*: If it is a standalone system, then it should support Identity Management for devices. However, if it is working together with people which should be capable of using it, then it should support the Identity Management service elements for both people and devices.
    - *Information Classification service*: As it will work with and/or create new information assets it will have to support the Information Classification service.
    - *Policy Management*: The system will have to support the policy management elements in order to be capable of creating, reading and understanding the (information access) policies.
    - *Meta/Information Management*: As the system will work with information assets, it will have to support these services.
    - *Audit support*: All systems should be auditable in order to check for the status and the security of the system.
- **COA framework Technologies:** The necessary support for COA framework Technologies will differ, depending on the type of the system and its purpose:
    - *Endpoint security*: When the system architecture does not comprise any endpoints, then it will not be necessary to implement endpoint security. However, as the Endpoint security service will comprise all endpoints in the future, one should plan support for it or just directly implement it.
    - *Encapsulation and Encryption support*: As the information assets will have to be processed, one should implement encryption and decryption support. However, encapsulation is only necessary for non-standalone systems or system components which exchange information.
    - *DRM related tool support*: If the system or components of the system work with external information assets, then they should support DRM related tools.

Other COA framework Technologies will only be necessary if the System Architecture comprises them such as VOIP, Mobile management, et cetera.

Next steps – Recommendations:

This section has only given a rough estimation of the elements which should be implemented. Which is not strange, knowing that there are many definitions and interpretations of the concept “system architecture” as defined in (Forum 2008e). That is why we recommend the following:

- **Clear definition of what can become a COA in terms of a systems architecture**: One should first further refine how to interpret the term System architecture and what kind of COA framework elements will be required by such a system architecture. This could lead to multiple classifications of a system and its architecture and therefore to multiple standards which describe the necessary COA framework elements.
- **Research how to implement the COA framework**: After the definitions have been clarified and one knows what exactly needs to be described and included in a system architecture, then one should research again which elements should be implemented and how that should be done.

- **Use roadmaps**: As most of the services and processes will have to be developed in line with the to be build roadmaps, one should define roadmaps for implementing the oncoming COA framework solutions inside the architecture.
- **Develop open standards:** In order to be capable of standardising and speeding up the COA framework implementation process, one should develop additional standards for COA system architecture development, to ease the implementation of the COA framework.

3.11.6. Summarising: the adoption of the COA framework

Looking back at this paragraph, we can summarise our findings as follows:

- **The COA framework and SaaS:** The COA framework delivers the means which are necessary to fulfil the needs of each actor in the discussed scenario. The landlords and the brokers will have to implement most of the COA framework elements, however this can vary for customers. Mutual trust relations will be possible and there will be transparency in how the information assets are handled and where they are. However, we have dealt only with one scenario and just enumerated the needs and the necessary COA framework elements, due to restraints in terms of time and resources. Additional research will be necessary for further detailing the current scenario and describe the other SaaS scenarios as well. Furthermore, one should create roadmaps and open inherently secure standards based on that research, which should accelerate the implementation of the COA framework in the SaaS environment.
- **The COA framework and an enterprise architecture:** We have defined what an enterprise architecture is and which additional actions will have to be taken to create a COA when one wants to use the TOGAF ADM. We have seen that all of the elements will be necessary and that they will have to be implemented in different phases of the ADM. However, additional research will be required for detailing the (implementation) processes and the creation of a data reference architecture, roadmaps and additional standards for TOGAF which should accelerate the implementation process.
- **The COA framework and a network architecture:** We have used a definition for defining the network architecture and have approached the network as a de-perimeterised network. As it is a de-perimeterised network, it should only include the COA framework Technologies, -Principles, -Attributes and the Risk Management processes. However, further research will be necessary to define and standardise the implementation process, roadmaps and additional standards for the network architecture itself.
- **The COA framework and a system architecture:** As a system architecture can vary a lot, depending on the system, its components and its purpose, we have tried to define the necessary components from the COA framework while allowing for a broader interpretation of the actual need of the elements. However, additional research will be necessary in order to: provide a more clear definition of a system architecture (or a classification of it), the necessary COA framework elements for that type or system architecture, the roadmaps to implement these elements, the implementation methodology for implementing them and a set of standards to standardise all of this, speeding up and easing the implementation process.

However, one could argue that in order to become a COA as an organisation, all of the information architectures will have to comply to the COA framework in order to make them completely ready for de-perimeterisation. This will enable one to use all of the COA framework contents inside the different architectures, with more elements in an Enterprise Architecture than in a Network Architecture.

3.12. Summary: The COA framework

All of the phases and, which have been introduced in section 3.1 (and listed in Figure 68 and Figure 69) have been covered and all of the research questions can be answered based on the finding of this chapter.
We can summarise this chapter, grouped by the research questions and other findings:

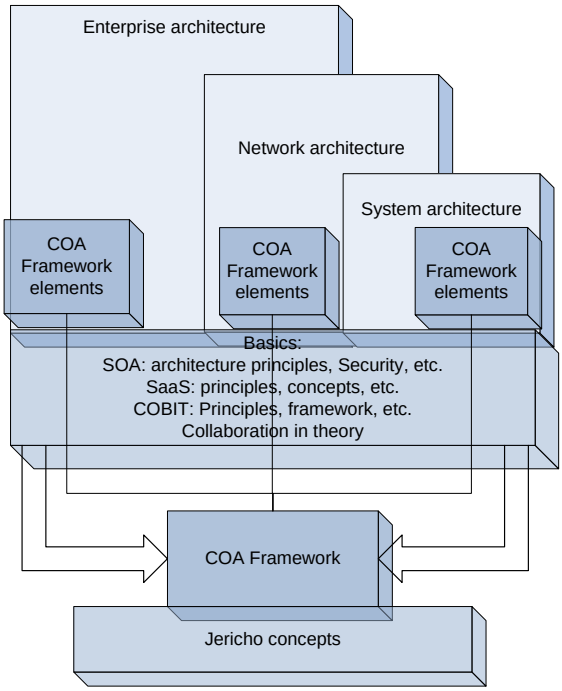


Figure 68: Thesis structure. COA related areas have been covered in this chapter.

- **The COA framework is**[151] the response to de-perimeterisation and consists of a set of resources which is aimed at transforming an architecture, which adopts the COA framework. The COA framework changes the focus of the architecture from the perimeter and the inside, to the outside. It re-perimeterises the architecture. This is done by providing the means for secure interconnectivity by increasing the emphasis at the COA framework Principles within a set of resources which will allow one to de-perimeterise. These resources (including the principles) are:
  - *Principles:* The following principles have been defined to change the focus of the architecture:
    - Participating parties: All components of a transaction chain must be known to the contracting parties at all of its endpoints. These endpoints have been chosen by collaborating parties and will be the responsibility of the owner of that endpoint.
    - Trust: One will have to understand the level of trust/confidence one will have to be transacting at, knowing the assets involved.
    - Assurance: One should agree to the level of trust/confidence he will be transacting at before the transaction itself starts and provide assurance that they will pertain at that level.
      - Compliance: One should comply to the rules and regulations of the security inside the collaborative group and audit one another to check if they do so. If one does not, he may be expelled.
      - Legal/regulatory/contractual: The collaborating parties must comply to applicable legal, regulatory, and contractual requirements and be able to resolve conflicts that may arise between these, through effective verification and enforcement mechanisms.
      - Privacy: One should handle privacy, thus PII with care.
      - Benefits and Obligations: As it is still in development, we have only found out that there is a set of additional obligations and requirements: Contractual obligations, service level agreements, customer expectations, corporate policy, and norms of good corporate citizenship are requirements that need to be aligned and implemented.

[151] Derived from research question one: "What is the Collaboration Oriented Architecture Framework?".

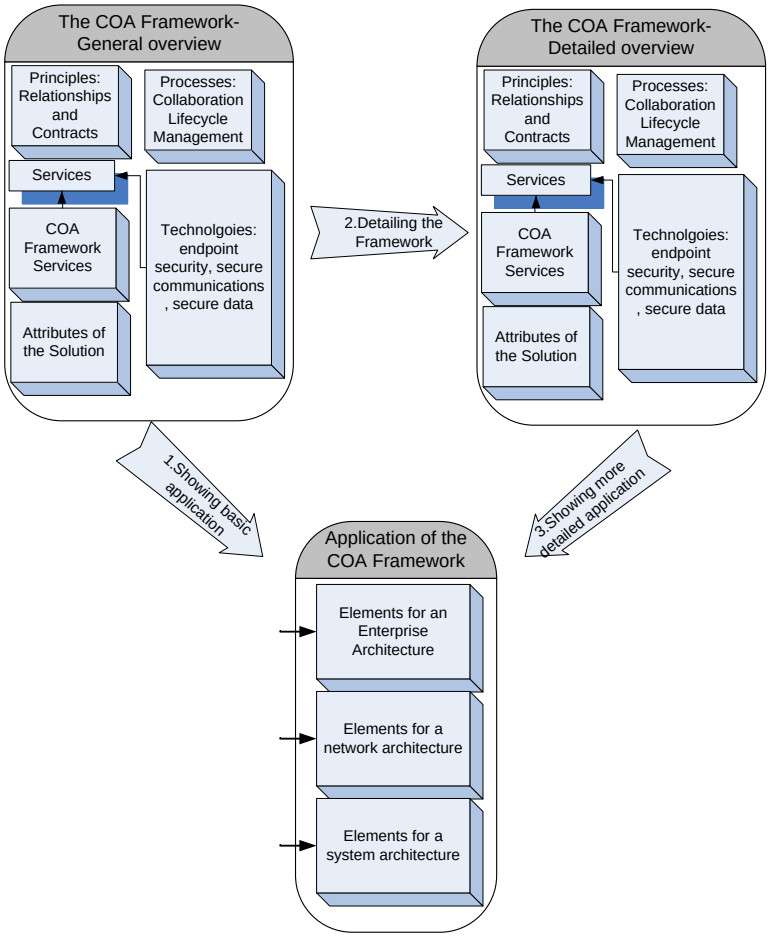


Figure 69: Processed steps of the thesis.

- All of these have been discussed and explained by working through several implications per principle.
- *Processes:* The COA framework provides a set of process which allows one to manage the complete lifecycle of people, risks, information assets, devices and enterprise relationships. The processes are designed in such a way that they will allow one to implement the benefits of Enterprise 2.0 in his collaborative environment. The processes or actually sets of processes that have been defined in the source documents were far from finished, that is why each process set has been further detailed and enriched with a set of requirements and recommendations based on the findings of chapter 2 and 3. The following process sets have been found:
  - People Lifecycle Management, which comprises the complete lifecycle of all personnel of the collaborative environment: from on-boarding, to off-boarding and everything in between such as monitoring and managing them. The process takes many things into account such as the identity, persona, capabilities, reputations and the impact of the entity. It relies the following services: Identity Management and Federation, Information Classification, Information Asset Management, Audit and Trust Management and Classification. We have defined a set of requirements for the processes based on the issues that will arise with the Identity Management, the audits, the privacy issues, the trust and – management and the involved Jericho Forum Commandments. We also defined a set of recommendations such as the creation of a privacy management system, the usage of OASIS standards, creating the services with a service based implementation, the creation of a process paper, the need for additional governance and most important, the need for further research of the processes.

- Risk Management, which is focussed on assessing, monitoring, managing and reducing the risks. The processes have been detailed as an elaboration and should be researched again and redefined to become a standardised part of the COA framework.
- Information Lifecycle Management, which manages the complete lifecycle of the information assets, from creation to destruction and everything in between such as access, reads, updates, monitoring and protection. It manages the information assets as well as their containers. It relies the following services: Identity Management, Information Classification, Audit, Policy Management and Trust Management and Classification. We have created a set of requirements for the processes based on Privacy, Policy Management, Data Classification, Data protection, Identity Management, Trust and – management, Endpoint security, Audit and the Jericho Forum Commandments. We also defined a set of recommendations such as the creation of a service based implementation, the implementation of reclassification steps in the process, the allowance for multiple classification methods, the development of the processes alongside the services, the creation of an open standard and a process paper and most important, the need for further research of the processes.
- Device Lifecycle Management, which will comprise the complete lifecycle of the devices, from the provisioning, till the de-provisioning and everything in between such as on-boarding, usage, off-boarding, management, monitoring and protection. These processes also comprise the software lifecycle management for now. They take several things into account such as the device identity, the endpoint security status and its capabilities. It relies the following services: Identity Management, Information Asset Management, Audit, Policy Management and Trust Management and Classification. We have created a set of requirements for the processes based on protocols and standards, policy management, data classification, privacy, data protection, Identity Management, trust and –management, endpoint security, audit, the Jericho Forum Commandments and a process paper. We also defined a set of recommendations such as assuring the compatibility with SOI, the creation of a service based implementation, the creation of a complete agent solution, the separation of the Device and Software Lifecycle Management processes, the creation of an open standard and a process paper and most important, the need for further research of the processes.
- Enterprise Relationship Management, which manages all of the aspects of the cross organisational relationships from monitoring the environment for new parties, to on-boarding, to the actual post-off-boarding processes. It manages the relationship, the usage of the relationships, the off-boarding of a collaborative relationship and everything surrounding these processes. It takes many things into account such as the environmental developments, the developments in the collaborative network, reputations, the value of the relationship and the trustworthiness of the organisations. It relies the following services: Identity Management, Information Asset Management, Audit, Policy Management and Trust Management and Classification. We have created a set of requirements for the processes based on collaboration, policy management, privacy, data classification, data protection, Identity Management, trust and /management, audit, the Jericho Forum Commandments and the process paper. We also defined a set of recommendations such as the creation for additional governance measures and frameworks to guide these processes, the creation of open standards and most important, the need for further research of the processes.

- *Services:* The COA framework also includes a set of services which will take care of many security issues. The following services have been defined as being part of the COA framework:
  - Identity Management, Federation and Reputation is a set of services which will manage the complete Identity Lifecycle from provisioning till de-provisioning and everything in between, including authorisation, management and monitoring of the identity. It handles the reputation, capabilities and more of the internal personnel. We have defined a set of requirements based on the findings of chapter 2 and 3. We also defined a set of recommendations such as: enabling the services by existing technologies,

implementing the solution in an open flexible Service Oriented fashion, separation of the services for personnel and device identification, the use of roadmaps to allow for user centric identity and most important, further research of available/required technologies and the creation of a set of open standards to standardise the service along its roadmap.

- Trust management and Information Classification: This comprises actually two sets of services. The first, the trust management service, will manage the external identities, devices and enterprises, the trust level of the external identities, devices and enterprises and later on interconnect to and check the trust broker and PII broker in order to be capable of managing that trust interconnected and to exchange the information assets with the trust broker such as legal/regulatory/contractual/ compliancy/contractual artefacts. The second, the classification service, will classify all of the information assets, identify PII and reclassify the data if necessary. The service will be capable of handling several types of classification methodologies and several types of classification models. Both of these service sets will have to collaborate with all of the other services by several means in order to allow for a good information exchange in which the assets are properly secured. We have defined a set of requirements based on the findings of chapter 2 and 3 for both of the services in general and for each service specific as well. We also defined a set of recommendations such as: separating these two services, the creation of a standalone Endpoint security service which will communicate with the trust management service, the usage the current technologies in order to make it a workable concept, the usage of a set of roadmaps to implement the necessary missing elements such as a trust broker and multiple types and methodologies for classification and most important, further research on this topic and the creation of a set of open standards to standardise the service along its roadmap.
- Policy Management is a set of services which will manage all of the information classification policies and apply them to the information assets. It will also provide information about them to other services and processes. Furthermore, it will manage the exchange of policies between two or more entities. We have defined a set of requirements based on the findings of chapter 2 and 3 for the service. We also defined a set of recommendations such as: the usage and research of current existing technology to make it a workable service, further research on the current service details, the creation of a set of roadmaps for developing many necessary means and the creation of a set of open standards to standardise the service along its roadmap.
- Information Asset Management (also called Meta/Information Management), which is a set of services that will manage and appropriately secure the information assets during their lifecycle. It will do so during creation, in storage, transit, while it is used and when it is destroyed. This also counts for the meta-information of the asset. The service works closely with the policy management service, the trust management service and the Identity Management service in order to apply policy decisions. We have defined a set of requirements based on the findings of chapter 2 and 3 for the service. We also defined a set of recommendations such as: the usage and research of current existing technology to make it a workable service, further research on the current service details, the creation of a set of roadmaps for developing many necessary means and the creation of a set of open standards to standardise the service along its roadmap.
- Audit: This set of services is focussed on auditing all of the services and processes by processing auditable events and controls. It will be designed to handle the audit scope and means as one should in a de-perimeterised environment. We have defined a set of requirements based on the findings of chapter 2 and 3 for the service. We also defined a set of recommendations such as: the creation of a set of services that allow for efficient computer aided audits, further research and the creation of a roadmap, comprising a set of open standards to make this a workable service.

- *(Quality) Attributes of the Solution:* The following quality attributes have been defined in order to measure if one has achieved his objective with the COA framework:

- Usability/manageability: The security measures should be easily understood and easily manageable, which can be tested by user test panels.
- Confidentiality: Confidential information should be protected at all times from unauthorised disclosure and the confidential network communications should be kept safe from eavesdropping and transparency.
- Integrity: Information assets should never be tampered with and always protected from attempts to do so. This can only be tested by accounting for every detected tampering with the information asset.
- Availability: The information assets should always remain available to the organisations which possesses them. This can be measured by using monitoring intelligent agents and the trust broker.
- Efficiency/Performance: The applied security should not have a negative impact on performance. This can only be measured by measuring the performance before and after the implementation of the framework, allowing one to see the differences.
- Effectiveness: Implementing the COA framework should create an effective approach to organizing and controlling secure data transport and storage among a wide range of existing and future corporate information systems. One can measure the ineffectiveness of the security and express the effectiveness as a percentage of 100 minus the ineffectiveness in terms of successful threats and attacks.
- Agility**:** Business agility alongside with flexibility should be accomplished by implementing the COA framework. How one can measure agility is of the scope of this thesis, yet the flexibility can be measured by the number (and granularity) of services, the change impact and the relative service independence.

  - *Technologies:* The COA framework also includes a set of technologies which are focused on three objectives and a group of "other technologies". The technologies will allow one to actually implement the services and the processes of the COA framework. The following technologies have been defined:
    - Those which provide end-point security: Several technologies will be used to manage the endpoint security or the trust state of the device.
    - Those which provide secure communications: Several technologies such as inherently secure protocols, WS standards, wireless security protocols, internet filtering and reporting, VoIP Security frameworks, Encryption based and related technologies to provide secure end to end communications.
    - Those which provide secure data: A set of technologies will be used to secure data. This will vary from DRM based tools to other SOA related technologies such as XACML and XKMS.
    - Other technologies: The Framework will have to incorporate other technologies as well such as Identity Management-tools, reputation systems and authentication related systems.

- **The importance of the COA framework**[152] can be defined of several points of view:
  - *The need of the COA framework in the current situation*: There are several issues at hand which could be resolved by the COA framework:
    - There are many security and management issues when two or more SOA based enterprises want to collaborate. Many solutions have been proposed, however, none of them have been really successful. The COA framework allows for a secure, safe and successful collaborative relationship, in which the security measures will be easy to manage, yet still effective.
    - There is an ever growing need for secure collaborations and until the COA framework, there is a lack of the appropriate measures to secure these collaborative endeavours.
    - The current collaboration models and relationships use many means to gain confidence in one another, which are often very time-consuming or easily forged.

[152] Derived from research question number two "Why is it important?".

  - The current collaborative relationships require a certain interconnectivity, intercompany IT-alignment and governance, which makes the current perimeterised approach flawed and allows the need for a de-perimeterised solution, as the COA framework, to grow.
  - There is a need for better information protection and especially for PII. The COA framework can fulfil that need and ensure that privacy matters will be handled with care.
  - The Jericho Forum has awakened the masses with their debate around de-perimeterisation. However, no solution has been provided until now. the COA framework is the first to tackle all of the issues surrounding de-perimeterisation.
- *Providing cross organisational secure Enterprise 2.0 benefits:* The COA framework will allow one to use the benefits of the Enterprise 2.0 concept by using SLATES across their collaborative environment. This will enhance the information exchange and provide a better ROI of the relationship.
- *Resolving trust issues in the SaaS environment:* The COA framework will allow the actors on the SaaS environment to get rid of the trust issues and gain a fully transparent collaborative relationship, in which trust, privacy and more aspects could be guaranteed. However, this benefit, could also be seen as a necessity.
- *Faster, more dynamic and still more manageable relationships:* The COA framework provides a set of processes and services which will allow one to enable relationships faster, more dynamic, and still have them manageable with a ease and transparency.
- *Enhancing support for internet business models:* All of the benefits above will provide better support for the internet business models of today, in terms of relationship management, asset exchange and security.

- **Adoption of the COA framework**[153] can be realised by implementing the COA framework elements into the information architectures. We have researched the following concepts on this field:
  - *COA framework in a SaaS environment:* We have used one of the three most popular SaaS scenarios and defined for each actor its needs and which elements of the COA framework should be implemented. However, more research will be necessary for actually implementing these elements. We have also recommended the usage of a set of roadmaps and the creation of a consumer COA framework.
  - *Adoption of the COA framework in an enterprise architecture:* We have defined the meaning of an enterprise architecture and used TOGAF ADM for showing which elements of the COA framework should be implemented in which phase of the creation of such an architecture. However, more research will be necessary for allowing one to actually implement the COA framework. We have recommended the creation of a set of detailed standards and a detailed implementation process and the development of a new reference data architecture for phase C in the TOGAF ADM.
  - *Adoption of the COA framework in a network architecture:* We have defined the meaning of a network architecture and from there on defined which elements would be necessary for a network architecture. However, one will still need to research the implementation process and create a set of standards which allows one to do so.
  - *Adoption of the COA framework in a system architecture:* We have defined the meaning of a system architecture and from there on defined which elements could be necessary for a system architecture. However, more research on this field will certainly be necessary, since a system architecture is to broad to be described given the resources that are available for this thesis.

[153] Derived from research question number three "How can an architecture for a system, a network and for an enterprise adopt the COA framework?".

- **The COA framework is based on** all of the Jericho Forum Commandments and many other aspects which have been described in paragraph 2.6, as we have defined in a mapping in paragraph 3.10.
- **COBIT and its use inside a COA:** We have also seen in section 3.6.6 that COBITs DS3 alone will not be sufficient for implementing governance in the Enterprise Relationship Management processes. That is why additional research on this field will certainly be necessary.

Now that we are at the end of this chapter, we have briefly answered all of the research questions. This will set the stage for the concluding chapter, chapter 4.

4. Conclusions and Recommendations

4.1.Introduction

Now that the research itself has been completed in chapter 2 and 3, we should be capable to answer all of the research questions that have been defined in section 1.4.2. By studying all of the basics and the concepts of the Jericho Forum in chapter 2, we gained the knowledge with which we defined the COA framework, its value and how it should be adopted by the three types of information architectures in chapter 3. See also Figure 70.

We finish this thesis with this concluding chapter with the following contents:

- The answers to the research question and its sub questions in paragraph 4.2.
- A set of recommendations based on our studies considering the COA framework and related elements in paragraph 4.3.
- A set of recommendations for further research derived from the findings of this thesis in paragraph 4.4.

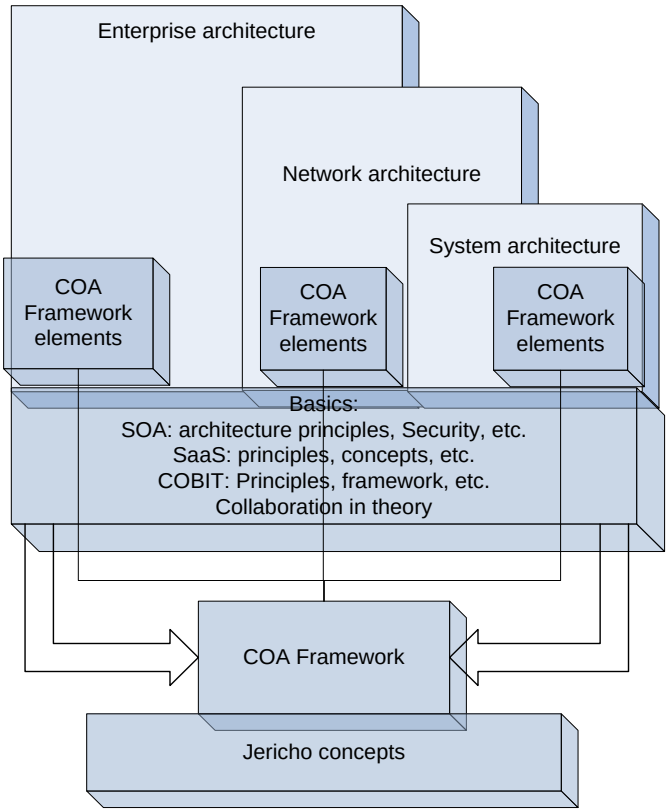


Figure 70: Thesis structure, transparent areas have been covered.

4.2.Conclusions

4.2.1.Introduction

We will briefly answer all of the research sub questions in this paragraph: question one in section 4.2.2, question two in section 4.2.3 and question three in section 4.2.4. Sections 4.2.2 and 4.2.4 will define the answer for our research question which will be answered in section 4.2.4.

4.2.2.The COA framework defined

The first sub question that was asked is:

What is the Collaboration Oriented Architecture Framework?

Which can be answered as follows:
The COA framework or Collaboration Oriented Architecture Framework is a set of principles, processes, services, quality attributes and technologies that will allow one to change the current information architecture into a Collaboration Oriented Architecture by adopting all of its components.

It is the architectural response to de-perimeterisation and allows one to re-perimeterise or de-perimeterise safely and secure.
The COA framework changes the focus of the traditional architecture from the inside with a perimeterised approach, to the outside by, focussing on interconnectivity by increasing the emphasis at the COA framework Principles. The COA framework also delivers a set of quality attributes (also called "Attributes of the Solution") which define some of the important attributes of the security measures that will be implemented when adopting the COA framework. These will show the architects and engineers what they will have to take in mind while designing and implementing the means that the Framework supplies.
The COA framework is designed to provide the newborn COA with the means to be capable of interconnecting in multiple ways as the business of today demands it. The COA can interconnect by multiple media with other organisations around the world and will be able to keep pace with the growing threats and the business need for faster and more flexible collaborative business arrangements.

In order to understand what the COA framework really is, one will have to understand its components as well. Even though they are still at development, we can already define them as follows:

- **The COA framework Principles:** the principles are designed to change the focus of the information architecture from a perimeterised approach to the inside, to an interconnectivity based approach to the outside. The following principles have been defined:
  - *Participating parties:* All components of a transaction chain must be known to the contracting parties at all of its endpoints. These endpoints have been chosen by collaborating parties and will be the responsibility of the owner of that endpoint.
  - *Trust:* One will have to understand the level of trust/confidence one will have to be transacting at, knowing the assets involved.
  - *Assurance:* One should agree to the level of trust/confidence he will be transacting at before the transaction itself starts and provide assurance that they will pertain at that level.

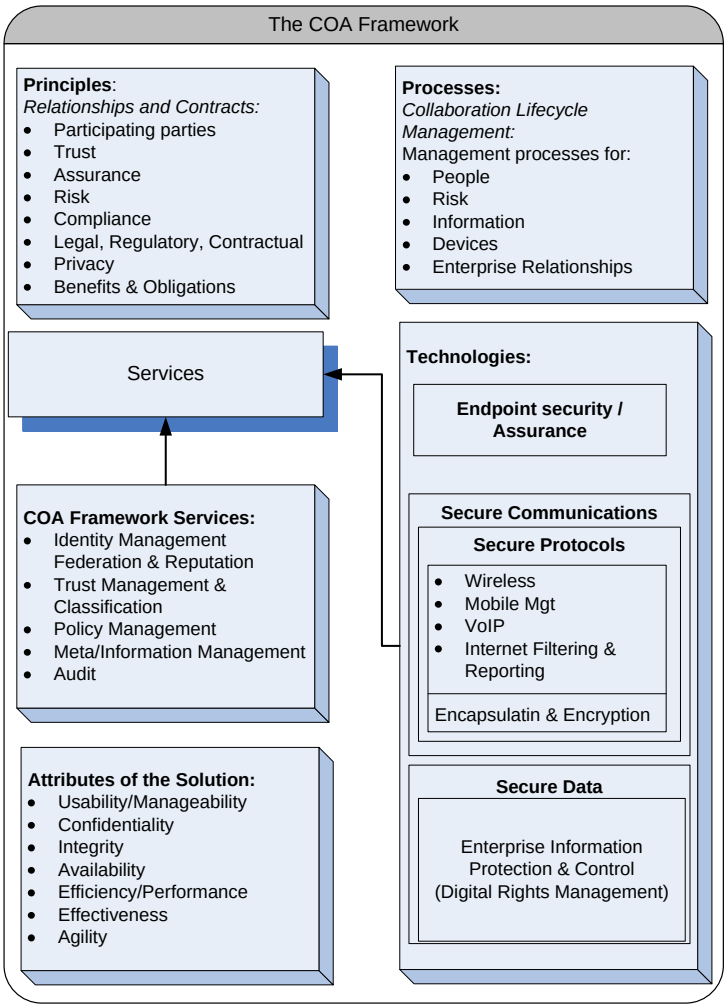


Figure 71: The Collaboration Oriented Architecture Framework from an architects' view (based on (Forum 2008e)).

- *Compliance:* One should comply to the rules and regulations of the security inside the collaborative group and audit one another to check if they do so. If one does not, he may be expelled.
- *Compliance*: One should comply to the rules and regulations of the security inside the collaborative group and audit one another to check if they do so. If one does not, he may be expelled.
- *Legal/regulatory/contractual*: The collaborating parties must comply to applicable legal, regulatory, and contractual requirements and be able to resolve conflicts that may arise between these, through effective verification and enforcement mechanisms.
- *Privacy*: One should handle privacy, thus PII with care.
- *Benefits and Obligations*: This principle is still in development, we have only found out that there is a set of additional obligations and requirements: Contractual obligations, service level agreements, customer expectations, corporate policy and norms of good corporate citizenship.

- **The COA framework Processes:** The processes that are delivered through the COA framework provide means to manage the lifecycle of people, risks, information assets, devices and enterprise relationships. The processes are designed in such a way that they will allow one to implement the benefits of Enterprise 2.0 in his collaborative environment. The following processes have been defined:
  - *People Lifecycle Management:* This is a set of processes which will manage the lifecycle of all of the personnel inside the collaborative environment. It starts with on-boarding the personnel and will manage and monitor them and at the end, off-board them again. The process takes many things into account such as the identity, persona, capabilities, reputations and the impact of the entity.
  - *Risk Management:* consists of a set of processes that is designed to assess, monitor, manage and reduce the risks.
  - *Information Lifecycle Management:* a set of processes that manages the complete lifecycle of information assets as well as their containers, from creation to destruction and everything in between such as access, reads, updates, monitoring and protection.
  - *Device Lifecycle Management*: This set of processes is responsible for managing comprise the complete lifecycle of the devices, from the provisioning, till the de-provisioning and everything in between such as on-boarding, usage, off-boarding, management, monitoring and protection. These processes also comprise the software lifecycle management for now. They take, while being executed, several things into account such as the device identity, the endpoint security status and its capabilities.
  - *Enterprise Relationship Management*: A set of processes that manages all the aspects of the cross organisational relationships such as the on-boarding, usage of the relationship and the off-boarding. It takes many things into account such as developments in the market and the collaborative network, reputations, the value of the relationship and the trustworthiness of the organisations.
- **The COA framework Services**: These services will take care of many (security) issues and will provide basic means for management, supporting the processes. The following services have been defined:
  - *Identity Management, Federation and Reputation*: actually a set of services which will manage the complete Identity Lifecycle from provisioning till de-provisioning and everything in between, including authorisation, management and monitoring of the identity. It handles the reputation, capabilities and more of the internal personnel.
  - *Trust management and Information Classification*: actually comprises two sets of services: The first, the trust management service, will manage the external identities, devices and enterprises, the trust level of the external identities, devices and enterprises and later on interconnect to and check the trust broker and PII broker in order to be capable of managing that trust interconnected and to exchange the information assets with the trust broker such as legal/regulatory/contractual/ compliancy/contractual artefacts. The second, the classification service, will classify all of the information assets, identify PII and

reclassify the data if necessary. The service will be capable of handling several types of classification methodologies and several types of classification models.
  - *Policy Management*: Again, a set of services. This set will manage all of the information classification policies, provide information about them and apply them to the information assets.
  - *Information Asset Management*: This set of services is also mentioned as Meta/Information Management. It manages and secures the information assets and their metadata during their lifecycle during creation, in storage, transit, use and when it is destroyed.
  - *Audit*: This set of services is focussed on auditing all of the services and processes by processing auditable events and controls.
- **The COA framework Quality Attributes or Attributes of the Solution:** The COA framework provides a set of quality attributes that can be used as measurements to check whether one has achieved its goal with the implementation of the Framework and as attributes to take in mind while implementing the COA framework. The following quality attributes have been defined:
  - *Usability/manageability*: The security measures should be easily understood.
  - *Confidentiality*: Confidential information should be protected at all times from unauthorised disclosure and the confidential network communications should be kept safe from eavesdropping and transparency.
  - *Integrity*: Information assets should never be tampered with and always protected from attempts to do so.
  - *Availability*: The information assets should always remain available to the organisations which possesses them.
  - *Efficiency/Performance*: The applied security should not have a negative impact on performance.
  - *Effectiveness*: Implementing the COA framework should create an effective approach to organizing and controlling secure data transport and storage among a wide range of existing and future corporate information systems.
  - *Agility***:** Business agility alongside with flexibility should be accomplished by implementing the COA framework.
- **The COA framework Technologies:** The COA framework also includes a set of technologies which are focused on three objectives and a group of "other technologies". The technologies will allow one to implement the services and the processes of the COA framework. The following technologies have been defined:
  - *Providing endpoint security*: Several technologies will be used to manage the endpoint security or the trust state of the device.
  - *Providing secure communications*: Several technologies such as inherently secure protocols, WS standards, wireless security protocols, internet filtering and reporting, VoIP Security frameworks, Encryption based and related technologies to provide secure end to end communications.
  - *Providing secure data*: A set of Digital Right Management related technologies will be used to secure data.
  - *Other technologies*: The Framework will incorporate other technologies as well such as Identity Management-tools, reputation systems and authentication related systems.

However, one should take in mind that the COA framework is far from finished. We have identified what the elements are and what still needs to be done to make it a workable solution. See also chapter 3, paragraphs 4.3, 4.4 and sections 4.2.4, 4.2.5 for a detailed coverage.

4.2.3. The importance of the COA framework

The second sub question that was asked is:

Why is it important?

Which can be answered as follows:
The COA framework is important because of the following reasons:

- **Because there is a need for it**: The COA framework can provide the following means that have become a necessity:
  - *The means for resolving security- and management issues in SOA collaborations:* There are multiple security management issues when two or more SOA based enterprises want to collaborate which have not been resolved easily.
  - *The means for secure collaboration:* There is a lack of the appropriate measures to secure many of the collaboration endeavours.
  - *The means for efficiently gain confidence in one another:* The current collaboration models and relationships use many means to gain confidence in one another, which are often very time-consuming or easily forged.
  - *The means for interconnectivity:* The current perimeterised approach is flawed and cannot provide the means for secure interconnectivity as the business of today requires it.
  - *The means for protecting information:* There is a growing need for better information protection and especially for PII.
  - *The means of resolving the issues of de-perimeterisation:* The Jericho Forum has awakened the masses with their debate around de-perimeterisation. However, no solution has been provided until now.
  - *The means of resolving the trust issues in the SaaS environment:* The SaaS market is facing many problems due to the lack of trust. If this can be resolved, then it could flourish in its full potential.
- **Because it delivers additional value:** The COA framework delivers the following additional value:
  - *Enterprise 2.0 benefits across the entire collaborative environment:* Implementing the COA framework will allow one to use the benefits of the Enterprise 2.0 concept by using SLATES across their collaborative environment. This will enhance the information exchange and provide a better ROI of the relationship.
  - *Fast, more dynamic and still manageable relationships:* The COA framework provides a set of processes and services which will allow one to enable relationships faster, more dynamic, and still have them manageable with a ease and transparency.
  - *Enhanced support for internet business models:* All of the benefits above will provide better support for the internet business models of today, in terms of relationship management, asset exchange and security.

See also the following paragraphs for a full coverage: 3.2 and 3.4.

4.2.4. The application or adoption of the COA framework

The third sub question that was asked is:

How can an architecture for a system, for a network and for an enterprise adopt the COA framework?

Which can only be partially answered, due to two reasons: First, the lack of completeness of the COA framework as noticed with the answer provided for the first question in section 4.2.2. There are no workable solutions defined for the services, processes and the requested technologies: all of them need further research and development. Second, there is a limited

amount of time and resources available to develop the workable solutions themselves and the necessary methodologies for implementing them into the architectures. However, the questions can be partially answered based on the current research results:

- **Adoption of the framework in a System Architecture:** Due to constraints in time and resources, no fitting methodology is found for implementing the COA framework into a System Architecture. By understanding that a System Architecture can comprise any system with any kind of purpose or even all systems with any purpose, defining the exact elements and their respective implementation is currently impossible. However, by using a broader definition, the reader himself can reason if the mentioned element would be a necessity in his systems architecture:
  - *The COA framework Principles and -Attributes:* The Principles and Attributes of the Solution should be used as design requirements for the system and its security. This will allow one to ensure that the architecture its implementation will be capable of surviving in a de-perimeterised environment, while delivering a maximum value in terms of safety, collaborative capabilities et cetera.
  - *The COA framework Processes:* The necessity of the COA framework Processes will differ per type of system and its goal. The architecture should adopt the Risk Management, Device Lifecycle Management and Information Lifecycle Management processes at all times. As for the People Lifecycle- and Enterprise Relationship Management processes: they should be supported by the architecture and implemented if they are in line with the goal of the system(s).
  - *The COA framework services***:** The necessity of the COA framework Services will differ per type of system and its goal. Most of the services should be included, however the type of necessary Identity Management support will differ based on the need to interact with personnel or just system agents. One will have to implement the full Identity Management Services set in case of the former, or just the device Identity Management measures in case of the latter.
  - *COA framework Technologies***:** The necessary support for COA framework Technologies will differ, depending on the type of the system and its purpose. As most of the systems will process information assets and will be interconnected, they will need security technologies such as encapsulation and encryption and DRM related tools. However, as it is not always the case that the architecture will comprise any end-points, the need for end-point security will not be evident until a certain moment. See also the recommendations in paragraph 4.3 for further details.
- **Adoption of the framework in a Network Architecture:** One should understand that a Network Architecture that is adopting the COA framework will change into de-perimeterised network. This answer lacks a methodology, again due to the lack of resources and time. However the necessary COA framework elements have been defined:
  - *The COA framework Principles and –Attributes*: The Principles and Attributes of the Solution should be used as principles for the network architecture. This allows one to create a de-perimeterised network which will be ready for the age of de-perimeterisation, provide QoS borders and proper means for collaboration.
  - *The COA framework Processes:* As most of the processes are service based and not network based, they will not be implemented in the Network Architecture. However the Risk Management process will have to be implemented: one will still need to assess the risks and threats at its network and take risk reducing actions for as far as possible.
  - *The COA framework Technologies:* As many of the COA framework technologies are actually part of the network infrastructure they will have to be included in the Network Architecture. This counts for Wireless/ Mobile management technologies, VoIP security frameworks, Internet Filtering and Reporting mechanisms, Encapsulation and encryption technologies and End-point security measures, as some devices of the Network architecture can become end-points.
- **Adoption of the framework in an Enterprise Architecture:** In order to adopt the COA framework with an Enterprise Architecture, one should use TOGAF its ADM. The ADM itself

delivers all the steps that are necessary to create an Enterprise Architecture. However, in order to implement the COA framework itself onto the Enterprise Architecture, the following additional actions were defined per phase of the ADM:

- *Additional actions for the Preliminary phase:* One should discuss / investigate the COA framework in this phase as this would be a good preparation for changing the focus of the architecture by using the COA framework principles.
- *Additional actions for Requirements Management:* One should think about the COA framework and its implications for governance and other concepts which are important in this phase.
- *Additional actions for Phase A:* First, Use the COA framework Principles as a part of the strategic goals and translate them into business requirements. Second, choose a set of COA framework Quality attributes and translate them into business requirements. Third, choose the basic needs of the COA framework: select those elements which can be implemented now and which could be implemented later. Fourth, take the aspects of (i.e. Jericho Forum Commandments) and drivers for de-perimeterisation and see if they can be linked to the current business drivers, to see how important the COA framework will be to the enterprise and its architecture.
- *Additional actions for Phase B:* Define an implementation of the COA framework Processes and Services: work through each process and service, define the organisational structure which will be necessary for them and additional the business goals and objectives, business functions, (sub-)services, processes and roles which are related to them. This automatically means that one will need to use roadmaps and see which standards of the processes can be used. Next, one will have to integrate these in such a way that they are becoming reusable building blocks which can be mapped to the organisational units. Try to estimate how many times they will be used and what information the building blocks will need. Again, one should adhere to the open standards in these estimations. From there on, define COA building blocks: identify those services which are based on the COA framework to be identified as COA building blocks, to show that the architecture entails to the COA framework. Apply also the details to them and see which standards will be necessary and if they can be based on old services from the current enterprise architecture which is in use. Finally: review the architecture with stakeholders in the collaborative environment, check if they are in line with the standards and the methodology used by the collaborative environment. The quality attributes can be used again in this phase for defining the service levels.
- *Additional actions for Phase C:* One should use the COA framework Services and Processes to define what data will be necessary to support the COA framework. One should use the Information Lifecycle Management processes and the Information Asset Management service as the primary data management model and set of building blocks. Next, one should ensure that selected Data Architecture Resources and Vision are in line with the principles of the COA framework and the prospects around de-perimeterisation. From there on, the COA building blocks can be identified while one should allow the collaborative environment to review them to check whether they chosen building blocks will be compatible with the ones used by partners. Next, one should use the applications of the COA Technologies to define which applications will be necessary, which are already in use and which need to be implemented according to their roadmaps and describe the further details as defined in (OpenGroup 2005). Again, one can use the COA quality attributes as quality attributes to specify service levels.
- *Additional actions for Phase D:* Use COA Technologies: one should further refine the to be used COA framework Technologies and apply them to the technology architecture as described in (OpenGroup 2005). The same goes for the services and processes of the COA framework.
- *Additional actions for Phase E:* Check for existing COA framework elements: check which elements of the COA framework already have been implemented according to the current valid standards based on the roadmaps. From there on check for those which still need to be build.

- *Additional actions for Phase F:* Create a roadmap based on the recommended roadmaps for the COA framework elements. Group them into projects which still need to be executed. Ensure that the roadmap is build according to the principles defined in (OpenGroup 2005). When finished, review the roadmap with the other stakeholders.
- *Additional actions for Phase G:* Detail the projects, show which are COA related. Check again for the impact on the organisation and the environment and use COA framework Principles and – Quality attributes of the Solution again to define acceptance criteria.
- *Additional action for Phase H:* Ensure that there is a set of change management processes that is compatible with the COA framework and the new scope due to de-perimeterisation.
-

See also paragraph 3.11 for a more detailed coverage.

4.2.5. The COA framework and its usage

A solid base for the research question has been made by answering the sub questions. All that is left for this paragraph is answering the research question itself:

"What is the Collaboration Oriented Architecture and how can it be used?"

As the question already has been answered in detail by the previous sections and even in greater detail in chapter 3, based on even more detailed research in chapter 2, the author does not want to repeat all of those details again. Instead, the research question itself will be answered by summarising the answers found in the previous sections of this paragraph:

The answer is, like the research question, twofold: first one know what the Collaboration Oriented Architecture is, which can be defined as follows:
The Collaboration Oriented Architecture Framework (COA framework) is the architectural response to de-perimeterisation. It consists of the following:

- **The COA framework Principles:** The principles are designed to change the focus of the information architecture from a perimeterised approach to the inside, to an interconnectivity based approach to the outside. They are focussed on knowing with who one transacting with, understanding the need of trust, ensuring that one will gain enough assurance from all of the parties that the necessary level of trust will be pertained, that risk is managed properly, that all parties will ensure compliancy, that they will respect privacy and follow the legal, regulatory and contractual obligations.
- **The COA framework Processes:** The processes that are delivered through the COA framework provide means to manage the lifecycle of people, risks, information assets, devices and enterprise relationships. The processes are designed in such a way that they will allow one to implement the benefits of Enterprise 2.0 in his collaborative environment.
- **The COA framework Services:** The services will hold the measures to de-perimeterise or re-perimeterise, while taking care of many security issues and creating support for the COA framework Processes. The following services have been defined: Identity Management, Federation and Reputation, Trust management and Information Classification, Policy Management, Information Asset Management and Audit.
- **The COA framework Attributes of the solution:** The COA framework provides a set of quality attributes that can be used as measurements to check whether one has achieved its goal with the implementation of the Framework. They will have to be taken in mind while implementing the COA framework. They consist of the following: the manageability/usability of security, the confidentiality, integrity and availability of the information assets, the efficiency and effectiveness of the security measures and the agility of the business.
- **The COA framework Technologies:** the technologies of the COA framework allow one to implement the COA framework Services and –Processes. They provide several means such as tools for end-point security, end to end secure communications and data protection.

Implementing the COA framework into one's information architectures will transform the perimeter based, internal focussed information architectures into Collaboration Oriented Architectures which are focused on the outside. This change of focus is realised by emphasising on the COA framework Principles.
The Attributes of the Solution will show the architects and engineers what they have to take in mind while designing and implementing the COA framework elements. Later on, those same statements can be used to measure if one has achieved its goal with the implementation.
The COA framework is designed to provide the newborn COA with the means to be capable of interconnecting in multiple ways and be as flexible as the business of today demands it.

The second part of the answer comprises how one can use the COA framework:
Using the COA framework, means implementing it into the information architectures. This can be done in various scenarios, especially within the information architectures of SOAs and SaaS using enterprises. The implementation of the COA framework into the information architectures will differ per type of information architecture. The Enterprise Architecture will have to comprise all of the COA framework contents, while the Network Architecture will do without many of the COA framework Services and Processes. However, if one wants his architecture to become a COA, then all of the information architectures should adopt the COA framework.

4.3.Recommendations

4.3.1.Introduction

Many recommendations have been made throughout chapter 2, where the domains coherent to the COA framework have been investigated and chapter 3, where the COA framework and its application has been defined.
As the focus of the thesis itself is on the COA framework, we will refrain from repeating all of the recommendations that have been found in chapter. They have been mentioned as to be "necessary knowledge" in order to study and define the COA framework in chapter 3. This does not mean that these recommendations from chapter 2 are not important, they should be taken in mind when one will further research the contents of the framework and its application.
The recommendations of chapter three will be summarised and grouped on a higher level between this and the next paragraph in terms of recommendations for further research (paragraph 4.4) and others (this paragraph) and on a lower level in terms of COA framework Processes, -Services, and the implementation of the Framework. Each of the group with its own section.

4.3.2.Recommendations related to the COA framework Processes

One should follow the following recommendations on the field of the COA framework Processes:

- **Common recommendations: T**he following recommendations account for all or most of the processes:
  - *The creation of a service based implementation:* The process should be implemented based on services, allowing one to take the full benefits of the available SOA.
  - *The creation of an open standard:* The processes should be standardised in an open and inherently secure standard, which is protocol and model independent and allows multiple vendors to come up with solutions that will be interoperable.
  - *The usage of roadmaps:* As the services, on which the processes will lean, will be further developed alongside roadmaps, so must the processes be accompanied by a set of roadmaps to give a clear picture of how they should evolve.

- *Secure communications and message protection between the processes:* The messages and the communications will have to be properly secured. Depending on the content of the messages, different types of protection will have to be taken in mind.
- *Privacy management system:* A management system should be designed with a PII broker to protect the PII and let the users be in control of that. Until it has been build, one should follow the OASIS standards in order to make it workable.
- *Additional governance:* An additional governance framework should be designed for auditing the processes and providing the right governance tools for the management of them.
- *Follow the requirements:* All of the process definitions have been accompanied by a set of requirements. These should be followed when designing or further researching the processes.

- Recommendations for People Lifecycle Management:
  - *Co-development with COA framework services:* In order to make sure that the processes are executable, one should develop them alongside their most important services such as the Identity Management and the Trust Management service.
  - *The creation of a process paper:* The Jericho Forum should come with a process paper that supports the development of the process.
- Recommendations for Information Lifecycle Management:
  - *Reading and access process should be separately implemented:* In order to allow partial blanking out of documents, the access and reading process should be separately implemented.
  - *The usage of risk taxonomy and traffic light protocol:* There should be an easy way of communicating risk of the assets and the classification level of the data. Both the risk taxonomy and traffic light protocol can add loads of value here.
  - *The creation of a process paper:* The Jericho Forum should come with a process paper that supports the development of the process.
- Recommendations for Device Lifecycle Management:
  - *Compatibility with SOI:* In order to make the processes compatible with the infrastructures of today, one should make sure that it can be easily adopted by companies that use SOI by making it compatible to SOI.
  - *Complete agent solution:* Vendors should try to create a security agent, which is capable of monitoring every named aspect of the device, and communicate the status to the user and the trust broker. The agent should be non-falsifiable to allow multiple types of end-point security architectures.
  - *Additional lifecycle management process:* The Software Lifecycle Management should be separated from the Device Lifecycle Management processes in order to allow a better research base and a more specific approach of the lifecycle.

4.3.3. Recommendations related to the COA framework Services

One should follow the following recommendations on the field of the COA framework Services:

- **Common recommendations:** The following recommendations account for all or most of the services:
  - *Use current existing technology:* To ensure that the services will be workable in the current situation, one should try to use currently existing technologies to create a first implementation of them, allowing for a workable solution.
  - *Implement the solution in an open flexible Service Oriented fashion:* To ensure that one can use several protocols, standards and other service specific entities, one should implement the services loosely coupled with clear in- and output definitions per service. This should ensure interoperability between the services, even when some of them are

being upgraded with new capabilities (i.e. User Centric approaches for Identity Management or automated classification services for the Classification Service).
  - *Create a set of inherently secure open standards:* To ensure that all of the implementations of the services along their roadmap will be universally exchangeable, one should create a set of inherently secure and open standards, which will allow one to follow all of the requirements and recommendations written in this document.
- Recommendations for Identity Management, Federation and Reputation:
  - *Separate services for devices- and personnel identification:* Some of the mechanisms for device Identity Management and authentication will differ from those for personnel. That is why one should implement those separately.
  - *Roadmap - Prepare for user centric identity:* There should be a roadmap for the Identity Management services that is focussed on implementing a globally accepted, fully accountable, user centric Identity Management system. One should ensure that there is enough flexibility in the current solution to implement the user centric identity solution.
  - *Ensure that privacy concerns will be met:* Privacy concerns should be met in design, build and testing of the service. This in order to guarantee both employees and customers that their privacy will be maintained. One should use other services such as Trust Management and Classification for classifying the information and later on use a PII broker, Policy Management to ensure the right information access policies, Meta/Information Management to protect the information and Audit to check the controls and events to ensure that the PII will be handled correctly.
  - *No biometrics:* The Identity Management system should not use biometrics as one of the default authentication mechanisms. This will fail as it can be easily compromised.
- Recommendations for Trust Management and Classification:
  - *Separate services:* In order to make the services more manageable, one should separate the classification service from the trust management services. Both of the services are focussed on different processes and aspects. This will remove some of the complexity and allow a more dynamic implementation of the services, since they will not be related so tightly anymore.
  - *Separate End-point security service:* The End-point Security service should be separated from the Trust Management services in order to reduce the complexity of the trust management services and make it more manageable.
  - *Roadmap – prepare for trust broker:* There should be a roadmap for the trust management services, which focuses on working with them to be developed into a trust management framework. It should first allow P2P interaction and federation, and later on work with a trust management framework and a trust broker. One should ensure that there is enough flexibility in the current solution and the solutions to come, so the trust management framework and PII broker can be implemented.
  - *Roadmap – prepare for multiple types of classification:* There should be a roadmap for enhancing the classification services, starting with non-automated classification, building towards advanced computer classification mechanisms that allow one to classify the ever growing amounts of information that is being created, updated and exchanged on a daily basis.
  - *Roadmap – prepare for end-point security for every device:* The end-point security service should no longer be focussed on an end-point. It should comprise all of the devices in order to give full transparency to the risks that are present in a collaborative environment.
- Recommendations for Policy Management:
  - *Understanding one's position in the collaborative network:* The service should be manageable in such a way that the manager can take the role of the organisation in the collaborative network into account. This will ensure that the manager will not take any actions that will create any discontent in the collaborative environment if the organisation is not in a commander role.

- *Roadmap – prepare for a fine-grained information access policy infrastructure:* There should be a roadmap for policy management that is focussed on implementing a globally accepted, fully accountable, very fine-grained information access policy infrastructure for policy management. One should ensure that there is also enough flexibility in the current solution to implement such an infrastructure.
- *Roadmap – prepare for a service that covers most types of policies:* There should be a roadmap for policy management, that focuses on implementing a globally accepted, policy management service that covers any of the fields where digital policies are used. One should ensure that there is also enough flexibility in the current solution to implement such a service.

- Recommendations for Meta/Information Management:
  - *Roadmap - Prepare for a fine-grained information infrastructure:* There should be a roadmap for the information management services that is focussed on implementing a globally accepted, fully accountable, fine-grained information infrastructure. One should ensure that there is enough flexibility in the current solution to implement such an infrastructure.
- Recommendations for Audit:
  - *Roadmap - Prepare for a new tactical and strategic scope of the audits:* There should be a roadmap for the audit services that is focussed on implementing a globally accepted, fully de-perimeterisation proof audit service.

4.3.4. Recommendations related to the COA framework application

The following recommendations have been derived from the issues that arose during the research on behalf of the implementation of the COA framework:

- **Recommendations for an application of the framework in a SaaS environment:** The COA framework is extremely suitable for resolving the trust issues in a SaaS environment as one can see in sections 3.11.2 and 4.2.3. This alone is worth the recommendation: "use the framework in the SaaS environment to resolve the trust issues". However, in order to make it fully successful, the following recommendations should be followed as well:
  - *Develop a consumer version of the COA framework:* As the COA framework is aimed at helping organisations, it is not focussed on assisting the individual consumer. It would be wise to assist all the actors in the environment and thus develop a consumer version of the COA framework. Allowing the consumer to make use of the business services, without having to go through many complexities, in an environment where all the other parties of the transaction chain have already adhered to the COA framework.
  - *Use the early recommended roadmaps to create a new one for the SaaS broker, landlord and consumers:* Many of the elements of the COA framework elements will require further research and development as one has seen in the earlier paragraphs (3.6 -3.9). Most of them should be developed according to a roadmap, allowing for a more sophisticated solution over time. These roadmaps can be used to develop a set of roadmaps for the players in this SaaS community, so that each player can evolve to a COA, allowing safe collaboration while resolving the SaaS trust issues.
  - *The creation of open and inherently secure standards***:** All of the products from the COA framework should be developed based on open inherently secure standards and protocols, allowing safe exchange of information (assets) and services between the parties in the SaaS community. This also counts for the implementation processes that will have to be developed to create a usable implementation methodology allowing the SaaS players to implement the COA framework.
- Recommendations on implementing the COA framework in an Enterprise Architecture:
  - *Create detailed standards***:** The standards for the processes and services will have to be detailed, in order to ensure a correct implementation in the enterprise architecture. The

standards considering the services should not just be detailed in terms of processes or service-building blocks, they should be detailed in terms of data and data flows as well. All of the standards of the recommended roadmaps should be detailed as max as possible, ensuring interoperability between the solutions adhering to these standards.
    - *Develop new reference data architectures:* A new reference data architecture set will have to be developed which takes the vast amounts and streams of data in mind which will come with de-perimeterisation. This will be necessary in order to support phase C of the ADM.
    - *Create additional standards to TOGAF for implementation:* The current TOGAF standard will have to be expanded with an additional COA standard in order to standardise the implementation processes of the COA framework elements.
- Recommendations on implementing the COA framework in a Network Architecture:
    - *Define the implementation process:* One will have to create a implementation process in order to implement the COA framework elements, allowing a network architecture to become a COA.
    - *Use roadmaps:* As some of the necessary technologies will still need further research and development, it would be wise to use roadmaps which allow for a systematic evolution of the network architecture and its components.
    - *Create a series of open and inherently secure standards:* In order to further accelerate and ease the adoption process of a COA network among organisations, one should develop a series of open standards that describe the COA framework elements and their implementation in a network architecture.
- Recommendations on implementing the COA framework in a System Architecture:
    - *Create a clear definition of what can become a COA in terms of a systems architecture:* One should first further refine how to interpret the term System architecture and what kind of COA framework elements will be required by such a system architecture. This could lead to multiple classifications of a system and its architecture and therefore to multiple standards which describe the necessary COA framework elements.
    - *Use roadmaps:* As most of the services and processes will have to be developed in line with the to be build roadmaps, one should define roadmaps for implementing the oncoming COA framework solutions inside the architecture.
    - *Develop open standards:* In order to be capable of standardising and speeding up the COA framework implementation process, one should develop additional standards for COA system architecture development, to ease the implementation of the COA framework.

4.4.Further research

As a research project is never finished, one will always have loose ends at the end of the research project. Even a single new (derived) fact can raise thousands of questions and directions to do more research in.
The same counts for this research project. As many concepts of- and surrounding the COA framework have been researched, many new questions have risen. We recommend the further research of the following:

- **Research Risk Management process**: these processes have only been elaborated on, without proper research, due to the lack of well defined sources and time. That is why this set of processes will have to be researched again when the respective sources are ready to use.
- **Further detail the COA framework Processes and – Services:** as the services and processes have been coarsely described, they will still need a further detailing research. One will have to research them again, this time with the quality attributes taken in mind, in order to define them with more details on both the conceptual level as well as the underlying implementation or physical levels. One should take the COA framework Attributes of the

Solution in mind while doing so, in order to ensure that the services and processes can be implemented according to those attributes.

- **Research differences between Server and Device Lifecycle Management in a COA:** as this aspect has stayed out of the scope of the thesis, it would be still quite interesting to see how they differ and how they both should be implemented in a COA.
- **Define measuring processes and detail them**: as the Attributes of the Solution have been partially detailed, they still lack a measuring process, which still needs to be defined and detailed.
- **Research relationships between the processes and services**: in order to get a better grip on the processes and services of the COA framework, one should check which relationships there are between processes, between services and between the processes and the services.
- **Research the applicability of other governance frameworks:** as we have seen that CObIT will not suffice for a de-perimeterised environment, other frameworks will have to be researched as well, to see whether there is a framework which will support the de-perimterised environment without having to apply time consuming changes.
- **Define and detail the application methodology of the COA framework to the information architectures:** as it was beyond the available resources to define the exact implementation methodology for applying the COA framework on the information architectures, one should further research them.
- **Research the technologies of the COA framework:** as it was beyond the available resources to define the exact technologies of the COA framework, one should further research them and check which technologies are still missing and how all of the necessary technologies should be implemented, grouped and managed.
- **Refine the trust broker and the PII broker:** as the thesis only provides some coarse details around these two concepts, one should research them more in order to make them workable solutions which can interact with the Trust Management service.

The final recommendation, one should not forget , is the following:
As we have seen in the Position Paper about the COA framework, many other elements will still have to be added and researched. Those will certainly remain a very interesting field for further research, besides those mentioned in the list above.

Literature

The literature list does not include the extra resources that have been used and noted in the footnotes considering different topics and organisations. It does also not include the extra resources used from the COA v2.0 release. See the footnotes and the COA v2.0 boxes for more details.

A.Araghi (2006). INFRASTRUCTURE FOR WEB SERVICES THAT PERFORM AUTOMATED REASONING. Queensgate, University of Huddersfield**:** 6.

Ahuja, G. (2000). "Collaboration networks, structural holes, and innovation: A longitudinal study." ABI/INFORM Global(45): 32.

al, D. C. L. e. (1985). Department of Defense Trusted Computer System Evaluation Criteria. Department of Defense Standard. Supersedes, United States Department of Defense**:** 116.

Alan, B., et all (2002). Using Service-Oriented Architecture and Component- Based Development to Build Web Service Applications, Rational Software Corporation**:** 16.

Amelia Maurizio, J. S., Peter Jones, Gail Corbitt and Lou Girolami (2008). Service Oriented Architecture: Challenges for Business and Academia. 41st Hawaii International Conference on System Sciences, Hawaii.

Anderson, R. (2008). Security Engineering, Wiley.

Area, S. S. T. O. G. I. M. W. (2004). Identity Management. O. Group. San Francisco, Open Group.

Arnold, J. (2008). "Position Paper End Point Security." JerichoForum: 3.

Barannikov, E. (2008). Jericho in depth, Authentication and accounting. Utrecht, Capgemini.

Bible (4000 BC). The fall of Jericho. The Book of Jehoshua.

Bingnan Xiao, W. T. T., Qian Huang, Yinong Chen, Ray A. Paul (2006a). SOA Collaboration Modeling, Analysis, and Simulation in PSML-C. IEEE International Conference on e-Business Engineering (ICEBE'06)**:** 8.

Bingnan Xiao, W. T. T., Qian Huang, Yinong Chen, Ray A. Paul (2006b). SOA Collaboration Modeling, Analysis, and Simulation in PSML-C. IEEE International Conference on e-Business Engineering (ICEBE'06).

Blake-Wilson, M. N., D. Hopwood, J. Mikkelsen, T. Wright (2006). rfc4366: Transport Layer Security (TLS) Extensions. N. W. Group, Internet Engineering Task Force.

Boonstra, A. (2002). ICT, mensen en organisaties, Pearson Education Benelux B.V.

Borisov, P. N. (2007). Introduction to Computer Security — Spring 2007. Illinois.

Bruning, A. (2008a). Trust broker Framework. Utrecht, Capgemini.

Bruning, A. (2008b). Trust broker Services. utrecht, capgemini.

C. Matthew MacKenzie, K. L., Francis McCabe, Peter F Brown, Rebekah Metz (2006). Reference Model for Service Oriented Architecture 1.0, OASIS**:** 31.

Cambridge, a. u. (2007). Cambridge Advanced Learner's Dictionarry. Cambridge, Cambridge.

Cameron, K. (2005). The laws of identity. Redmond, Microsoft**:** 12.

Chaffey, D. (2004). E-Business and E-commerce Management: Strategy, implementation and practice. Harlow, Pearson Education limited.

Chou, W. (2002). "Inside SSL: The Secure Sockets Layer Protocol." IT Pro(July | August 2002): 6.

Clark, K. P. (2008). AUTOMATED SECURITY CLASSIFICATION. Exact Sciences. Amsterdam, Vrije Universiteit. **Master:** 79.

David L. Cannon, T. S. B., Brady Pamplin (2006). CISA Certified Information Systems Auditor - Study Guide. Indianapolis, Indiana, Wiley Publishing.

David Lacey, J. A., John Walsh (2006). "Statement Paper Regulation, Compliance & Certification." JerichoForum: 3.

David Sprott, L. W. (2003). "Understanding SOA." CBDIJournal.

Demarteau, A. (2008). Requirements for a Human-Centric Trust Management System in an Open De-Perimeterised Network Environment. Dept. of Information and Computing Sciences. Utrecht, Utrecht University. **Doctoral:** 83.

Dirk Hanenberg, F. A. (2008). Feasibility of Fully ASP-based Organizations - Case Study: ASPublished Publishing House. Industry and Information Science. Tilburg, Avans Hogeschool. **Bachelors:** 116.
Earl, T. (2007). "What is SOA." Retrieved 01-04-08, 2008, from http://www.whatissoa.com/p10.asp.
Erl, T. (2005). Service-Oriented Architecture: Concepts, Technology, and Design, Prentice Hall PTR.
Farber, D. (2006). "The invasion of Enterprise 2.0 software." Retrieved 22-10-2008, from http://blogs.zdnet.com/BTL/?p=3898.
Forum, J. (2005). "Visioning White Paper What is Jericho Forum?" JerichoForum: 41.
Forum, J. (2006a). "Position Paper - VoIP in a de-perimeterised world." JerichoForum: 4.
Forum, J. (2006b). "Position Paper "Enterprise Information Protection & Control" (Digital Rights Management)." JerichoForum: 4.
Forum, J. (2006c). "Position Paper Federated Identity." JerichoForum: 2.
Forum, J. (2006d). "Position Paper Internet Filtering & Reporting." JerichoForum: 5.
Forum, J. (2006e). "Position Paper The Need for Inherently Secure Protocols." JerichoForum.
Forum, J. (2006f). "Position Paper Trust and Co-operation." JerichoForum: 4.
Forum, J. (2006g). "Position Paper Wireless in a de-perimeterised world." JerichoForum: 2.
Forum, J. (2007a). "Jericho Forum Commandments." JerichoForum: 2.
Forum, J. (2007b). "position paper data/information management." JerichoForum: 4.
Forum, J. (2007c). "Position Paper Information Access Policy Management." JerichoForum: 3.
Forum, J. (2007d). "Position Paper IT Audit in a De-perimeterised Environment." JerichoForum: 4.
Forum, J. (2007e). "Position Paper Principles for Managing Data Privacy." JerichoForum: 5.
Forum, J. (2007f). "White Paper Business rationale for de-perimeterisation." JerichoForum: 4.
Forum, J. (2008a). "COA Process Paper Device Lifecycle Management." JerichoForum: 4.
Forum, J. (2008b). "COA Process Paper Enterprise Relationship Management." JerichoForum: 3.
Forum, J. (2008c). "Jericho Forum." Retrieved 17 March 2008, 2008, from https://www.opengroup.org/jericho/index.htm.
Forum, J. (2008d). "The Need for Inherently Secure Communications." JerichoForum: 4.
Forum, J. (2008e) "Position Paper Collaboration Oriented Architecture." JerichoForum **Volume**, 6 DOI:
Foster, I. (2008) "Service-Oriented Science." **Volume**, DOI:
Fox (2008). Risk Taxonomy - Technical standard. Berkshire, Open Group.
Gambetta, D. (1988). Can we trust trust? Trust: Making and Breaking Cooperative Relations. D. G. (Ed.). New York**:** 213-237.
Gilpin, M. (2005). Topic Overview: Service-Oriented Architecture, Q3 2005. Cambridge, Forrester**:** 7.
Giudice, D. L. (2004). Service Oriented Architecture:
The Foundation for Digital Business. Forrester**:** 16.
Glenn Hanson, B. I., Evie Gray, Darren Smith, Dave Sutherland, Keith Junker. (1994, 18-11-08). "Federal standard -1037C, Glossary of Telecommunication Terms ", from http://www.its.bldrdoc.gov/projects/1037c/1037.html.
GODFROIJ, A. J. A. (1981a). Dynamische netwerken.
GODFROIJ, A. J. A. (1981b) "Sociale actiesystemen / interorganisationele relaties." **Volume**, DOI:
Hardt, D. (2005). Identity 2.0, SxIP.
Heasley, J. (2005). Securing XML Data. InfoSecCD Conference. Kennesaw, GA, USA, ACM**:** 3.
Heffner, R. (2005a). The Elements Of SOA Maturity- Major Checklist Categories For Developing Your SOA Strategy. Trends. Cambridge, Forrester**:** 7.
Heffner, R. (2005b). Your Strategic SOA Platform Vision Cambridge, Forrester**:** 27.
Heffner, R. (2007). Embedded SOA Management Solutions. Cambridge, Forrester**:** 16.
Henry Mintzberg, D. D., Frances Westley, Jan Jorgensenb (1996). "Some Surprising Things About Collaboration- Knowin a How People Connect Ma es It Wor& Better." organisational dynamics(spring 1996): 12.

Henry Peyret, A. P. (2005). an overview for capgemini.
Henry S. Teng, R. R., Martin Jordan (2008). "Position Paper IT Audit in a Deperimeterized Environment." JerichoForum: 5.
Herwig, B. M. S. V. J. G. V. W. M. (2008). "ALIGNING TECHNOLOGY WITH BUSINESS AN ANALYSIS OF THE IMPACT OF SOA ON OUTSOURCING." Journal of Theoretical and Applied Information Technology: 9.
Hinchcliffe, D. (2007). "The state of Enterprise 2.0." Retrieved 21-10-2008, 2008, from http://blogs.zdnet.com/Hinchcliffe/?p=143.
Hurwitz, J. (2006) "Thinking from Reuse - SOA for Renewable Business." **Volume**, 13 DOI:
Hutinski, S. G. a. Ž. (2007a). Standard Based Service-Oriented Security. foi.hir, Faculty of Organization and Informatics, University of Zagreb
8.
Hutinski, S. G. a. Ž. (2007b). Standard Based Service-Oriented Security. Zagreb, Faculty of Organization and Informatics - University of Zagreb**:** 8.
Institute, I. G. (2007a). COBIT 4.1. Framework Control Objectives Management Guidelines Maturity Models. Rolling Meadows, IT Governance Institute.
Institute, I. G. (2007b). Cobit 4.1 Excerpt. Executive summary Framework. Rolling Meadows.
Ismail Khriss, E. L., Guy Tremblay, André Jacques (2007). Towards Adaptability Support in Collaborative Business Processes. 2008 International MCETECH Conference on e-Technologies, IEEE.
Ivar Jørstad, S. D., Do Van Thanh (2005). A Service Oriented Architecture Framework for Collaborative
Services. the 14th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprise (WETICE'05), IEEE.
J. Schlyter, W. G. (2006). rfc4255: Using DNS to Securely Publish Secure Shell (SSH) Key Fingerprints. N. W. Group, The Internet Society.
James Joshi, A. G., Walid G. Aref, Eugene H. Spafford (2001). Digital Government Security Infrastructure Design Challenges. West Lafayette, Center for Education and Research in
Information Assurance and Security
Purdue University**:** 8.
James McGovern, S. W. A., Michael E. Stevens, James Linn, Vikas Sharan, Elias K. Jo (2003). A Practical Guide to Enterprise Architecture, Prentice Hall PTR.
Jason Hogg, D. S., Fred Chong, Dwayne Taylor, Lonnie Wall, Paul Slater (2005). Web Service Security. Scenarios, Patterns, and Implementation Guidance for Web Services Enhancements (WSE) 3.0.
Jerichoforum (2007). "Position Paper Network Security & Quality of Service." JerichoForum: 2.
John Wack, K. C., jamie Pole (2002). Guidelines on Firewalls and Firewall Policy-Recommendations of the National Institute of Standards and Technology. Gaithersburg,, Computer Security Division
Information Technology Laboratory
National Institute of Standards and Technology**:** 74.
joint technical comittee ISO/IEC JTC 1, I. t., Subcommeittee SC 27, IT Security Techniques (2005). ISO/IEC 17799 Information technology-Security techniques-Code of practice for information security management. Winterthur, ISO en IEC**:** 130.
Jones, J. A. (2008). An Introduction to Factor Analysis of Information Risk**:** 79.
Jørstad, e. I. (2005). "A Service Oriented Architecture Framework for Collaborative Services." IEEE(Proceedings of the 14th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprise): 5.
Jothy Rosenberg, D. L. R. (2004). Securing Web Services with WS-Security, Sams Publishing.
Joziasse, S. (2008). Strategic control in a deconstructed firm. International Business. Rotterdam, Erasmus university. **masters:** 100.
Knippel, R. (2005). Serivce Oriented Enterprise Architecture. IT univeristy. Copenhagen, IT-University. **Master MSC:** 125.
Krafzik, D. e. a. (2005). Enterprise SOA – Service-Oriented Architecture Best Practices.
Leijden, F. v. d. (2008). TrustNOG AAN TE VULLEN!

Liam O'Brien, P. B., Jon Gray (2008). "Business Transformation to SOA: Aspects of the Migration and Performance and QoS Issues." SDSOA.
MacKinnon, W. G., G. Cray, D. (2008). Enterprise Information Systems and Strategic Flexibility. Hawaii International Conference on System Sciences, Hawaii, IEEE.
Maddock, B. (2004). Port Knocking: An Overview of Concepts, Issues and Implementations. SANS GIAC GSEC Practical. B. Maddock**:** 12.
Mamoon Yunus, R. M. (2005). WISE 2005, Springer-Verlag Berlin Heidelberg.
Mark Chang, J. H., Enrique Castro-Leon (2006). Service-Orientation in the Computing Infrastructure. the Second IEEE International Symposium on Service-Oriented System Engineering (SOSE'06).
Markku Sääksjärvi, A. L., Henry Nordström (2005). "EVALUATING THE SOFTWARE AS A SERVICE BUSINESS MODEL: FROM CPU TIME-SHARING TO ONLINE INNOVATION SHARING." IADIS International Conference e-Society: 9.
Marle, C. v. (2007). The Internet Business Models' relations to the Jericho Security Architecture. Technische Informatica. Utrecht, Hogeschool Utrecht. **Bachelor:** 54.
Massuthe, P. (2005). Operating Guidelines - an Automata-Theoretic Foundation for the Service-Oriented Architecture. Proceedings of the Fifth International Conference on Quality Software, IEEE.
Mattessich, P. W., M. Murray-Close, et al. (2001). Collaboration: what makes it work, Amherst H. Wilder Foundation.
McAfee, P. A. (2006). "Enterprise 2.0, version 2.0." Retrieved 22-10-2008, from http://blog.hbs.edu/faculty/amcafee/index.php/faculty_amcafee_v3/enterprise_20_version_20/.
Measham, J. (2001). Value Less Security. Chesterfield, Consignia Research Group.
Metsaars, P. (2008a). Software as a Service Security Architecture. Network Infrastructure Design. Heerlen, Hogeschool Zuyd. **bachelor**.
Metsaars, P. (2008b). Software as a Service Security Architecture. Network Infrastructure Design. Heerlen, Zuyd University. **Bachelor:** 66.
Michael Menzel, I. T., Christian Wolter, Christoph Meinel (2007). Proc. Stuttgarter Softwaretechnik Forum- SOA Security -Secure Cross-Organizational Service Composition. Stuttgart, Fraunhofer IRB-Verlag.
Mike Craig, D. K. (2007). BANK OF AMERICA CRMS – SOA STRATEGY / GOVERNANCE PROJECT. Soa guidance book v.02.
Natis, Y. V., et all (2006). Predicts 2007: SOA Advances, Gartner**:** 14.
Nelly Delessy-Gassant, E. B. F., Saeed Rajput, and Maria M. Larrondo-Petrie Patterns for Application Firewalls. Boca Raton, Dept. of Computer Science and Engineering
Florida Atlantic University**:** 19.
Nickul, D. (2007) "Service Oriented Architecture (SOA) and Specialized Messaging Patterns." **Volume**, 15 DOI:
Nicolas Gold, A. M., Claire Knight, Malcolm Munro (2004). "Understanding Service- Oriented Software." IEEE Software(april 2004): 71-77.
Nooteboom, B. (2005). "Learning to Trust." Multidisciplinairy economics: 65-81.
Ohrstrom, J. (2007). Business Value using Service Oriented Architecture. Industrial Information and Control systems, KTH. Stockholm, KTH. **Master:** 175.
OpenGroup, t. (2005). TOGAF. Version 8.1 Enterprise Eidtion. Berkshire, The Open Group.
Oppliger, R. (1998). "Security at the Internet Layer." IEEE: 5.
Peterson, G. (2005). "Service Oriented Security Architecture." Information Security Bulletin **volume 10**: 6.
Pezzini, M. (2005). Applied SOA: Best Practices From the Best Practitioners. Gartner Symposium ITXPO, Cannes, Gartner.
Philip Kotler, D. C. J., and Suvit Maesincee (2002). Marketing Moves: A New Approach to Profits, Growth, and Renewal, Harvard Business School Press.
Poppensieker, M. (2006). Requirements of Service-Oriented Architecture. Wirtschaftsinformatik 2. Trier, uni. **Lehrstuhl:** 110.

Ralph Welborn, V. K. a. f. b. S. B. (2008). The Jericho Principle. How companies use strategic collaboration of value to find new sources of value, Wiley.
Randy Heffner, L. F., Alex Cullen, Craig Symons, Christine E. Atwood (2006). The Scope And Focus Of SOA Governance. Client Choice topic. Cambridge, Forrester.
Rob High, J., Stephen Kinder, Steve Graham (2005). IBM's SOA Foundation - An Architectural Introduction and Overview, IBM Business Consulting Services: 68.
Robinson, R. (2004). "Understand Enterprise Service Bus scenarios and solutions in service-oriented architecture, Part 1." Retrieved 6-18-2004, 2004, from http://www-106.ibm.com/developerworks/library/ws-esbscen/.
Rogelio Aguilar Alamilla, e. a. (2008). CISM Review Manual 2008. Rolling Meadows, ISACA.
Rogers, S. (2005). Business Forces Driving Adoption of Service Oriented Architecture POSTPVAVERDER! IDC, IDC: 24.
S. Lehtinen, C. L. (2006). rfc4250: The Secure Shell (SSH) Protocol Assigned Numbers. N. W. Group, Internet Society.
S. Santesson, A. M., J. Ball (2006). rfc4346: TLS User Mapping Extension. N. W. Group, Internet Engineering TaskForce.
Sabater, J., & Sierra, C. (2005). "Review on Computational Trust and Reputation Models. ." Artificial Intelligence Review: 33-60.
Sam Weber, P. A., Michael McIntosh (2007). A Framework for Multi-Platform SOA Security Analyses. IEEE International Conference on Web Services (ICWS 2007), IEE.
Sanjeev Kumar, V. D., and M. S. Krishnan (2007). Does SOA Improve the Supply Chain? An Empirical Analysis of the Impact of SOA Adoption on Electronic Supply Chain Performance. the 40th Hawaii International Conference on System Sciences - 2007, Hawaii, IEEE.
Santesson, S. (2006). rfc4680: TLS Handshake Message for Supplemental Data. N. W. Group, Internet Engineering TaskForce.
Sarbanes, O. (2002). Sarbanes-Oxley Act of 2002. Congress. City of Washington, findlaw: 66.
Schneier, B. (2003). Beyond Fear. New York, Copernicus Books.
Seccombe, A. (2007). Trust, Jericho Forum.
Services, I. G. T. (2007). Infrastructure considerations for service-oriented architecture., IBM: 36.
Simons, P. (2006). "Statement Paper Encapsulation & Encryption." JerichoForum.
Sluiter, J. (2006). Service-Oriented Architecture and Deperimeterisation. Serivce-Oriented Architecture | The way we see it. London, Capgemini: 20.
Sprott, D. (2005). Business Flexibility through SOA, CBDI Forum: 19.
Stan, A. (2008a). Jericho in depth, Secure Communications. Utrecht, Capgemini.
Stan, A. (2008b). Jericho in depth... The road to Jericho. The Master Series. M. Plas. Utrecht, Capgemini.
Standardization, I. O. f. (2005). Information technology – Security techniques – Information security management systems – Requirements. ISO/IEC 27001:2005(E),.
Stanton, R. (2005). "Inside out security: de-perimeterisation." Network Security **2005**(4): 4-6.
Surekha, D., et all (2006). SOA Practitioners' Guide (part 1, 2, 3), BEA Systems: 18, 52, 57.
Surya Nepal, J. Z. (2008). A Conflict Neighbouring Negotiation Algorithm for Resource Services in
Dynamic Collaborations. 2008 IEEE International Conference on Services Computing.
T. Dierks, C. A. (1999). rfc2246: The TLS Protocol Version 1.0. N. W. Group, Internet Engineering TaskForce.
T. Dierks, E. R. (2006). rfc4346 - The Transport Layer Security (TLS) Protocol Version 1.1. Request for Comments. N. W. Group, Internet Engineering TaskForce.
T. Ylonen, C. L. (2006a). rfc4251: The Secure Shell (SSH) Protocol Architecture. N. W. Group, The Internet Society.
T. Ylonen, C. L. (2006b). rfc4252: The Secure Shell (SSH) Authentication Protocol. N. W. Group, Internet Society.
T. Ylonen, C. L. (2006c). rfc4253: The Secure Shell (SSH) Transport Layer Protocol. N. W. Group, The Internet Society.

T. Ylonen, C. L. (2006d). rfc4254: The Secure Shell (SSH) Connection Protocol. N. W. Group, The Internet Society.
Tanenbaum, A. S. (2003). Computer Networks, Fourth edition, Prentice Hall.
Teheux, L. (2008). Jericho in depth, Authorization & Endpoint Security. Utrecht, Capgemini.
Trieloff, C. (2005). "Open for Integration." Enterprise Open Source Journal: 2.
Tsai, W. T. (2005). "Service-Oriented System Engineering: A New Paradigm." Service-Oriented System Engineering, 2005. SOSE 2005. IEEE International Workshop: 3-8.
Tsourveloudis NC, V. K. (2002). "On the Measurement of Enterprise Agility." Journal of Intelligent & Robotic Systems **33**: 329-342.
Vanhanen, T. (2003). REQUIREMENTS AND A FRAMEWORK FOR BROKER BASED INTEGRATION IN SERVICE-ORIENTED ARCHITECTURE. Department of Computer Science and Information Systems. Jyväskylä, University of Jyväskylä. **Master:** 136.
W3C. (2004). "World Wide Web Consortium: Web Services Glossary." Retrieved 25-03-08, 2008, from http://www.w3.org/TR/ws-gloss/.
W. T. Tsai, M. M., Yinong Chen, Farokh Bastani (2006). Perspectives on Service-Oriented Computing and Service-Oriented System Engineering. of the Second IEEE International Symposium on Service-Oriented System Engineering, IEEE Computer Society.
W. T. Tsai, Q. H., Bingnan Xiao, Yinong Chen and Xinyu Zhou (2007). Collaboration Policy Generation in Dynamic Collaborative SOA. Eighth International Symposium on Autonomous Decentralized Systems (ISADS'07), IEE**:** 8.
W.T.T sai, Q. H., Jingj ingX u, YinongC hen, Ray Paul (2007). Ontology-based Dynamic Process Collaboration in Service-Oriented Architecture IEEE International Conference on Service-Oriented Computing and Applications(SOCA'07), IEE**:** 8.
X. Zhou, W. T. T., X. Wei, Y. Chen, B. Xiao (2006). Pi4SOA: A Policy Infrastructure for Verification and Control of Service Collaboration. IEEE International Conference on e-Business Engineering (ICEBE'06), IEE**:** 8.
Y. Natis, R. S. (2003). Introduction to Service-Oriented Architecture. Gartner, Gartner**:** 6.
Ying Huang, S. K., Jen-Yao Chung (2004). A Service Management Framework for Service-Oriented Enterprises. IEEE International Conference on E-Commerce Technology.
Yuri Demchenko, L. G., Cees de Laat, Bas Oudenaarde (2005). Web Services and Grid Security Vulnerabilities and Threats Analysis and Model, IEEE**:** 6.

Appendices

Appendix A1: The 11 Jericho Commandments

Highlights, based on version 1.2 (Forum 2007a) :

Fundamentals:

1. The scope and level of protection should be specific and appropriate to the asset at risk.
2. Security mechanisms must be pervasive, simple, scalable and easy to manage.
3. Assume context at your peril.

Surviving in a Hostile World:

4. Devices and applications must communicate using open, secure protocols.
5. All devices must be capable of maintaining their security policy on an un-trusted network.

The need for trust:

6. All people, processes, technology must have declared and transparent levels of trust for any transaction to take place.
7. Mutual trust assurance levels must be determinable.

Identity, Management and Federation:

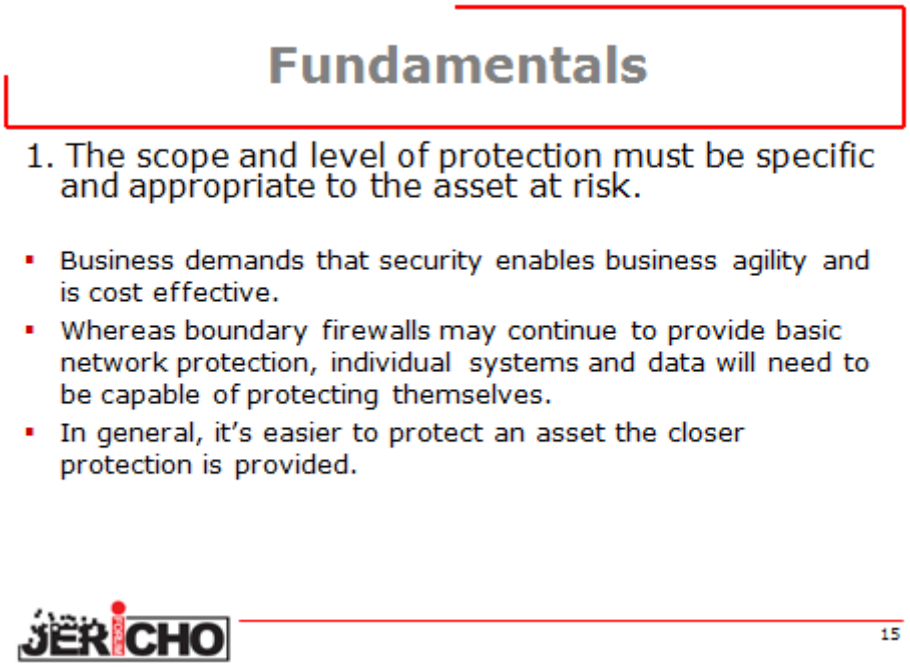
8. Authentication, authorisation and accountability must interoperate / exchange outside of your locus / area of control.

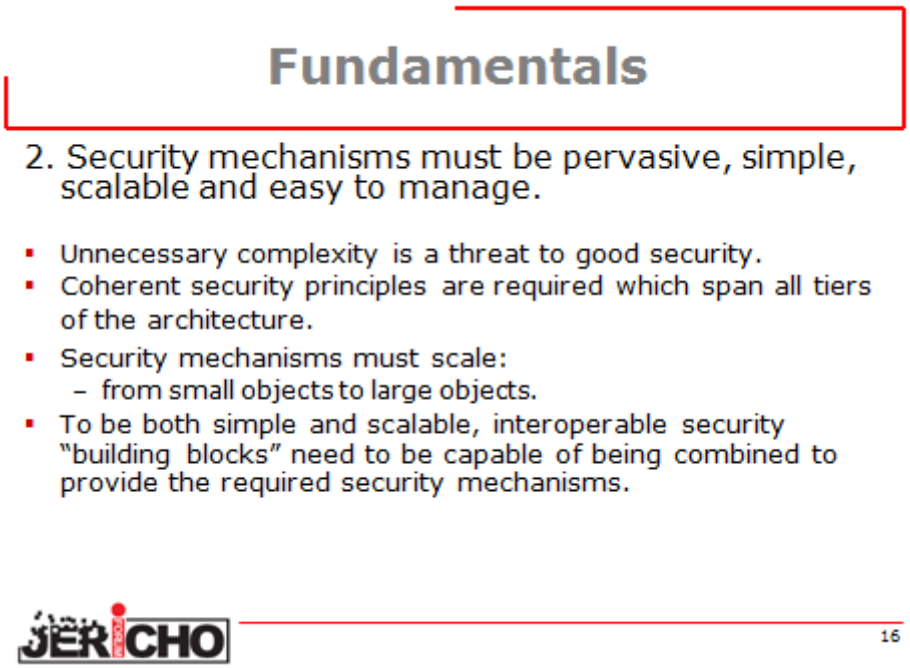
Access to data:

9. Access to data should be controlled by security attributes of the data itself.
10. Data privacy (and security of any asset of sufficiently high value) requires segregation of duties/privileges.
11. By default, data must be appropriately secured when stored, in transit and in use.

Appendix A2: Jericho Commandments explained

This appendix holds a part of the he presentation "IT Audit and Identity Management Challenges in a De-Perimeterisation Scenario" by Henry S. Teng. CISSP, CISM, given on the 2nd Annual Identity Management Summit 2007 By MIS Training.
They explain the Jericho Forum Commandments a little more:





Fundamentals

3. Assume context at your peril.

- Security solutions designed for one environment may not be transferable to work in another:
  - thus it is important to understand the limitations of any security solution.
- Problems, limitations and issues can come from a variety of sources, including:
  - Geographic
  - Legal
  - Technical
  - Acceptability of risk, etc.

JERICHO

17

Surviving in a hostile world

4. Devices and applications must communicate using open, secure protocols.

- Security through obscurity is a flawed assumption
  - secure protocols demand open peer review to provide robust assessment and thus wide acceptance and use.
- The security requirements of confidentiality, integrity and availability (reliability) should be assessed and built in to protocols as appropriate, not added on.
- Encrypted encapsulation should only be used when appropriate and does not solve everything.

JERICHO

18

Surviving in a hostile world

5. All devices must be capable of maintaining their security policy on an untrusted network.

- A "security policy" defines the rules with regard to the protection of the asset.
- Rules must be complete with respect to an arbitrary context.
- Any implementation must be capable of surviving on the raw Internet, e.g., will not break on any input.

JERICHO

19

The need for trust

6. All people, processes, technology must have declared and transparent levels of trust for any transaction to take place.

- There must be clarity of expectation with all parties understanding the levels of trust.
- Trust models must encompass people/organisations and devices/infrastructure.
- Trust level may vary by location, transaction type, user role and transactional risk.

JERICHO

20

The need for trust

7. Mutual trust assurance levels must be determinable.

- Devices and users must be capable of appropriate levels of (mutual) authentication for accessing systems and data.
- Authentication and authorisation frameworks must support the trust model.

JERICHO

21

Identity, Management and Federation

8. Authentication, authorisation and accountability must interoperate/ exchange outside of your locus/ area of control.

- People/systems must be able to manage permissions of resources they don't control.
- There must be capability of trusting an organisation, which can authenticate individuals or groups, thus eliminating the need to create separate identities.
- In principle, only one instance of person / system / identity may exist, but privacy necessitates the support for multiple instances, or once instance with multiple facets.
- Systems must be able to pass on security credentials/assertions.
- Multiple loci (areas) of control must be supported.

JERICHO

25

Finally, access to data

9. Access to data should be controlled by security attributes of the data itself.

- Attributes can be held within the data (DRM/Metadata) or could be a separate system.
- Access / security could be implemented by encryption.
- Some data may have "public, non-confidential" attributes.
- Access and access rights have a temporal component.

22

Finally, access to data

10. Data privacy (and security of any asset of sufficiently high value) requires a segregation of duties/privileges

- Permissions, keys, privileges etc. must ultimately fall under independent control
  - or there will always be a weakest link at the top of the chain of trust.
- Administrator access must also be subject to these controls.

23

Windesheim

Finally, access to data

11. By default, data must be appropriately secured both in storage and in transit.

- Removing the default must be a conscious act.
- High security should not be enforced for everything:
  - "appropriate" implies varying levels with potentially some data not secured at all.

24

Appendix A3: Jericho Roadmap

Based on (Forum 2007f), url:
http://www.opengroup.org/jericho/Business_Case_for_DP_v1.0.pdf , visited at 04-07-2008