Authentication and accounting

The Bachelors Series

Jericho in depth…

Authentication and accounting

Governing collaboration in Jericho networks

Evgeny Barannikov

Capgemini's Security & Innovation Research Centre, based in the Netherlands, focuses on near future IT Security solutions. The Jericho forum's vision on network de-perimeterization and Boundaryless Information flow™ has been the starting point for this research centre. Research papers from this centre appear in two distinct categories;

The Master Series
Researcher holding a masters degree in Informatics or are in the process of obtaining a master degree publish in the Master Series. The participating University and the Capgemini Security & Innovation Research centre have approved publications in this category.
Publications in this Series for 2008;

- Jericho in depth… Secure Communications by A. Stan
- Jericho in depth… The road to Jericho by A. Stan

Planned publications in this Series for 2008;

- Demystifying trust by F. van Leijden
- Jericho in depth… Automated Security Classification by K. Clark
- Jericho in depth… Trust Management for Trust brokers by A. Demarteau

The Bachelor Series
Researchers holding a bachelors degree in Informatics or in the process of obtaining a bachelors degree publish in the bachelor series. Their University and the Capgemini Security & Innovation Research centre have approved publications in this category.
Publication in this series for 2008;

- Jericho in depth… Endpoint security by L. Teheux
- Jericho in depth… Authentication and Accounting by E. Barannikov
- Jericho in depth… Trust broker Services by A. Bruning
- Jericho in depth… Trust broker framework by A. Bruning

Planned publications in this series for 2008;

- Jericho in depth… Controlling the COA framework by J. Willemsen
- Jericho in depth… Fully ASP based by D. Hanenberg & F. Aardoom

Preface

Since the dawn of the Internet at the ending of 1969 a lot has changed, I'm sure nobody will disagree with a statement like that. During the last couple of years however, we seem to have hit a mid-life crisis of the Internet. The sudden boost of Internet technology over the past decade does not fit well with our outdated design principles for network security. Most organizations hold tight to their fortress approach in trying to protect the internal network from the hostile Internet. Understandable, but not really realistic. In the Netherlands, we are particularly proud of our water management techniques. In a country that lays for more then sixty percent below sea level we know that we have to build and maintain solid dikes to prevent our country from flooding. Having holes in these dikes quickly diminishes the whole purpose of have a dike. The same holds true for perimeter defence in computer networks. Information leakage via email, hyves, my space or mobile data solutions like iPod or USB diminishes the purpose of perimeter security. Today's business world is one of collaboration, one of working together., one of global markets. The Internet is the ideal candidate to support this collaboration. The Jericho Forum (Open Group), formed by Security professionals from the largest organisations in the world described their vision of network de-perimeterization and boundryless Information flow™ in various publications. These visions formed the starting point for Capgemini's Security & Innovation Research Centre.

Together with the best universities in the Netherlands, Capgemini's offers academic researchers and graduate students to ability to conduct empirical academic research into the topic of Collaboration Oriented Architectures or to conduct feasibility studies into the Jericho Forums visions.

Marco Plas

Head of Jericho Research
Capgemini Security & Innovation Research Centre
Capgemini Netherlands

Abstract

The primary goal of the Information Technology is to support the enterprise's processes and not to limit them. The demand in collaboration is much higher today than a few years ago. Ability to provide services or to outsource processes is essential for an organisation. Closed security domains do not allow free information exchange between the organisations. New architecture is required which will allow boundaryless information flow. Jericho Forum which is a subdivision of the Open Forum has introduced the concept of de-perimeterisation. The forum published a number of the white papers which describe this new architecture. Being a platinum member of the Jericho Forum, Capgemini has gathered a research team which consists of the students from the Dutch universities. The aim of the research is to analyse the de-perimeterisation concept and define possible solutions for this new type of architecture.

The research is divided into 5 interdependent domains: AAA framework, Trust broker, encryption, endpoint security and data classification. AAA framework research consists of authentication, authorisation and accounting. All together they cover 7 processes. Each of them is assigned to a researcher. The first and the last one are covered in this research paper. The research is phased into general and specific parts. In the first phase the general architecture and interconnectivity of the processes are defined. Inputs and outputs of all the processes are determined so that interactions can be mapped. In the second phase the research is narrowed down to authentication and accounting. Authentication and accounting processes in the de-perimeterised network must be able to extend outside the perimeter of the network and exchange information with other security domains. There are few emerging technologies that allow this. The research comparatively investigates these technologies and recommends the most suitable one. The principles and methods of communication we rely upon in our real life have been established for years and they manage to successfully accommodate us in all our needs. The best way to create a safe collaborative environment in the Internet is to translate these principles and apply them to the digital reality. The emerging technologies bring this idea closer to implementation. The research results of this thesis serve as building blocks of the new simple and secure architecture which will eventually replace the current one. Network de-perimeterisation enables effortless collaboration and creates a cost effective environment for organisations and individuals.

Contents

Introduction

The limitations of our current IT architecture become more obvious with every day.
We observe that a modern enterprise network is not ready to accept a tremendous amount of external connections. Excessive perimeters make it operate in inefficient way. Organisations are surrounded by the virtual borders which do not allow any extension outside the perimeter. Enterprise boundaries in the real world are gradually dissolving as it is understandable that without collaboration no single enterprise can survive. A trend to focus on a single process and outsource the rest to somebody who can solve a problem professionally and for a lower cost is not completely reflected in the digital world. The trend there is completely the opposite - deploy as many firewalls as you can if you want to survive. Being the subdivision of the Open Group, Jericho Forum was the first one to ring the bell. They have published a number of white and vision papers describing the concept of the ideal network which is open and secure at the same time. By publishing their position papers and defining the standards they introduce the idea of de-perimeterisation. Unfortunately most of those concepts stayed on paper and haven't found their way to materialise. A trend of building a digital fortress in the hostile environment is understandable. Information is an important asset of the enterprises and it may not be compromised. Standing on the crossroads between collaboration and safe secure network we are trying to find the right way. Why not choose the middle way as it is promoted by Jericho Forum? Many questions arise. How can one safely collaborate in this digital hostile world? Is it possible at all? It is understandable that a new architecture is required which can support business needs and which can deploy right technical means to move the collaboration seamlessly forward. Being a platinum member of the Open Group, Capgemini has decided to undertake the challenge to build the new architecture which is based on the concept of de-perimeterisation. In December of 2006 they have gathered a research group which consists of the students of the Dutch universities. The aim of the research is to analyse the position and vision papers and find the possible solutions which will be secure and open. Being the leader of the research team Drs. Marco Plas has defined the five distinct domains of the research: AAA framework, Trust broker, encryption, endpoint security and data classification. Each of them is assigned to a researcher.

The AAA framework is one of the corner stones of the new architecture. Trust enables collaboration and AAA framework ensures consistency and security. The main difference with the "classic" framework is that it can be extended to other domains. In the de-perimeterised network user identities can be taken outside the perimeter but one can still control user activities and hold them accountable. Authentication process must be able to establish user identity regardless of its origination domain. Authorisation process analyses the authentication assertions, additional parameter and decides if they can satisfy the security requirements.
If sufficient data is collected and trustworthiness is established it authorises the user to access the data. Accounting process tracks user activities. In de-perimeterised network it must be able to exchange audit data between domains and monitor the security violations.

Identity, authorisation and auditing data exchange are positioned centrally in this new architecture. Networks have evolved from the stand alone computers to the independent security domains, where all devices can communicate in a free and secure manner. Now it is time to evolve further and let the security domains communicate with each other, just like the local computers do.

The evolution process is inevitable; Jericho Forum and our research group are the catalysts in this process. Jericho Forum establishes principles and standards, which form the foundation of the new architecture. We develop it further and search for the suitable technologies, which become the building stones of the new collaborative Internet driven world.

Jericho Project Research Group

For Jericho Forum Project[1], the security strategy is built on defining and implementing new approaches for trust management, AAA framework, end point security, data classification, and secure communications in order to provide security services such as privacy, authentication, non-repudiation, integrity in a de-perimeterised network. These topics will be addressed in detail in the research conducted for Jericho Forum Project. Jericho Forum is a consortium of the Open Group. It is an international IT security thought-leadership group dedicated to defining ways to deliver effective IT security solutions that will match the increasing business demands for secureIT operations in our open, Internet-driven, globally networked world.[2] The Open Group is a vendor-neutral and technology-neutral consortium, whose vision of Boundaryless Information Flow™ will enable access to integrated information, within and among enterprises, based on open standards and global interoperability.[3] Both of these two organisations have published multiple white papers on boundaryless information flow and de-perimeterisation. These papers define standards a de-perimeterised network must comply with.

The goal of Jericho Project is to develop new security architecture and design approach that will enable business to grow safely and securely in an open, Internet-driven, networked world. In this new security architecture, each device is capable to protect itself and each asset of the network is individually protected. De-perimeterisation requires security to be at the heart of the organisation's distributed technology architecture. Security has to be implemented in end-user devices, application services, and it has to efficiently and effectively protect organisations' critical information assets themselves. The aim of Jericho Project Research Group is to finally define a roadmap for the implementation of this new defined security architecture.

Five researchers under the direct supervision of Drs. Marco Plas compose the research group for Jericho Forum. The work within the research group is directed towards a comparative research of possible methods, models, technologies, cryptographic algorithms

[1] This text is partially general. It is written together by all members of the research team and may be included in every thesis.

[2] http://www.opengroup.org/jericho/

[3] http://www.opengroup.org/

and security protocols from the network security area. The goal of the research within Jericho Forum Project is to provide possible solutions for the new security architecture in de-perimeterised networks.
The research conducted for Jericho Project Research Group is divided into five distinct parts that focus on the following inter-connected topics of research:

- AAA Framework
- Trust broker
- Endpoint security
- Data classification
- End-to end encryption

Each of these domains is assigned to a researcher, who specialises in one or two subjects.

Eleven commandments

From the number of the white paper there is one, which summarises all the principles of the network de-perimeterisation. This paper is called Jericho Forum Commandments. There are eleven fundamental principles which should serve as a benchmark when defining a new architecture.

Fundamentals

1. **The scope and level of protection should be specific & appropriate to the asset at risk**
   It is easier to protect an asset by moving the protection closer.
2. **Security mechanisms must be pervasive, simple, scalable & easy to manage**
   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.
3. **Assume context at your peril**
   Every security solution has its own limitation. It is important to understand it. One may not easily transfer a solution to another environment where it is not designed for.

Surviving in a hostile world

4. **Devices and applications must communicate using open, secure protocols**
   One should use simple, open and secure protocols for communication.
5. **All devices must be capable of maintaining their security policy on an untrusted network**
   A security device placed in hostile environment must be able to protect itself and maintain its security policy.

The need for trust

6. **All people, processes, technology must have declared and transparent levels of trust for any transaction to take place**
   Trust should govern every transaction. It may vary depending on location, attributes, authentication context, etc.
7. **Mutual trust assurance levels must be determinable**
   Entities must be able to establish a trust level of another entity they communicate with.

Identity, Management and Federation

8. **Authentication, authorisation and accountability must interoperate exchange outside of your locus / area of control**
   AAA framework must be able to extend outside the security domain. Assertions issued by one authority must be accepted by another, eliminating by this the creation of multiple identity instances.

Access to data

9. **Access to data should be controlled by security attributes of the data itself**
   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.
10. **Data privacy (and security of any asset of sufficiently high value) requires a segregation of duties/privileges**
    Segregation of duties is required for the securing of the high value assets.
11. **By default, data must be appropriately secured when stored, in transit and in use**
    Data can exist in three states: being stored, being transferred and being used. Each of these states has own security risks. Security of the states must be appropriate to the data value.

1. Jericho network core components

The eleven commandments from the Jericho Forum can be used as design principles for a network architecture. After defining the principles that where derived from the commandments, our research group defined the requirements for the core Jericho service components and interlinked them. These service components are described in more detail further in this chapter.

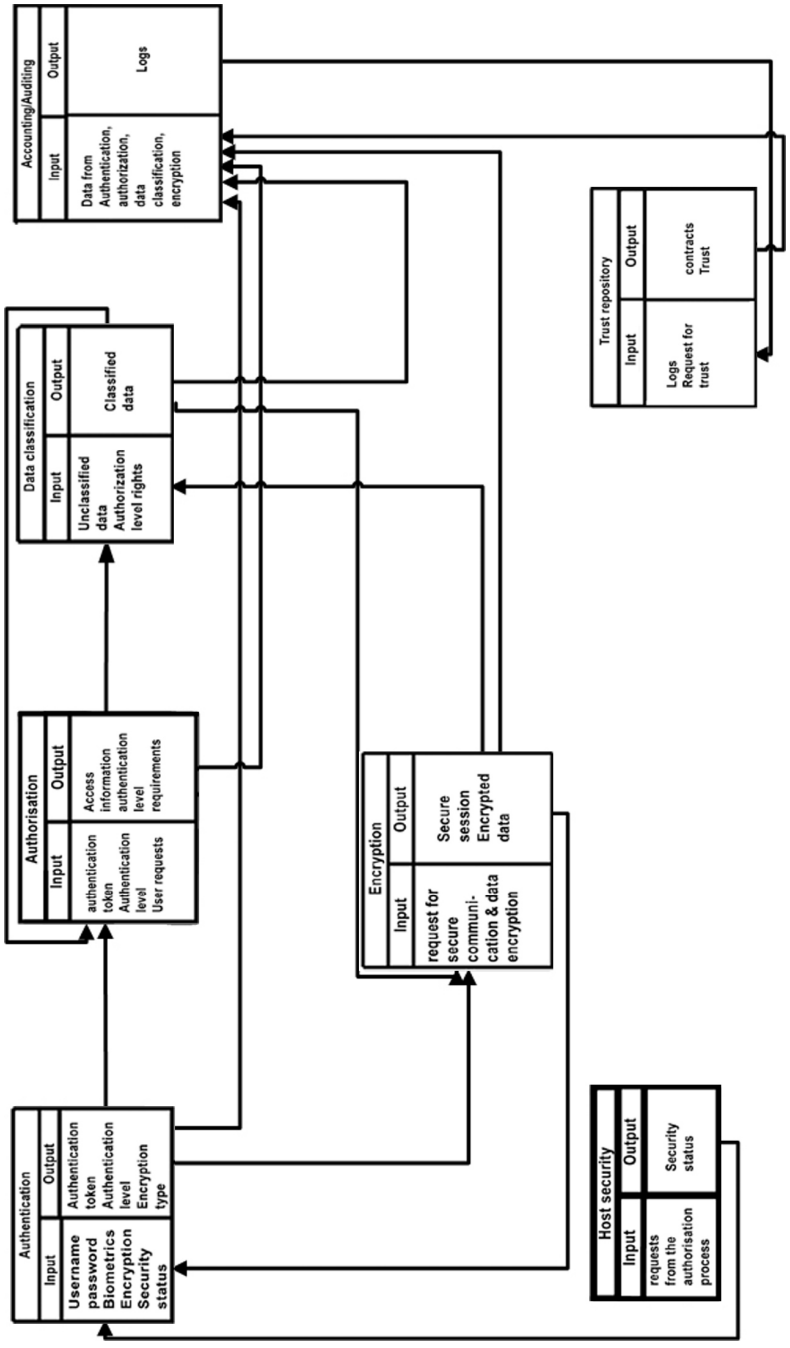


Figure 1 – Jericho Forum architecture process scheme

End-to-end encryption

Within the scope of Jericho Forum Project, the goal of end-to-end encryption[4]research is to investigate the possibilities offered by cryptography for designing and implementing suitable security protocols within Jericho networks for achieving secure communications. The starting point for this research is based on Jericho Forum Commandment[2] number 4 that states the following: "Devices and applications must communicate using open, secure protocols".

The research on end-to-end encryption for secure communications aims to offer recommendations for Jericho networks regarding the following aspects:

- Establishing the requirements for secure communications within Jericho networks
- Investigating a range of security protocols that can be used for end-to-end encryption for Jericho Project
- Choosing the most adequate cryptographic algorithms and primitives, in terms of security offered and performance, that should be used for security protocols that offer end-to-end encryption in Jericho networks
- Defining a roadmap for the implementation of the proposed solutions for Jericho networks

In order to provide end-to-end encryption for the data in transit, in the context of Jericho Project, a number of steps have to be followed.
First, the entities involved in the communication have to be authenticated in a handshake protocol. Second, there will be established a secure connection between the entities, and a secure session will be set up for transmitting the content securely. This step is being performed or not, according to the output of authentication, authorisation processes. Then, the adequate cryptographic primitives are chosen for achieving, further in the communication process, certain security services (e.g. confidentiality, integrity, authentication of the source of messages, non-repudiation). Also, the most suitable ways of distributing the keys are chosen and an agreement is reached. Typically, the public-key algorithms are used for secure key exchange, while the symmetric-key algorithms are used for encrypting the data in transit. Moreover, for the research on end-to-end encryption in the context of Jericho Forum Project, the sources of the transmitted messages are authenticated, so the aim is to achieve authenticated encryption for data in transit.

In Jericho Project, the end-to-end encryption research topic is inter-connected with the following modules that are part of Jericho Project research also: data classification, Trust broker, accounting and authentication modules. The data is previously classified before encryption occurs. So, certain security protocols and/or cryptographic primitives may be given preference for implementing end-to-end encryption depending on the type of data

[4] Excerpt from the book "Secure Communications" by A. Stan, Capgemini 2008

(e.g. highly sensitive, confidential, public etc.). In addition, entities have to authenticate before transferring data. Moreover, logs of the authentication process, of the encryption process will be kept for further checking. The Trust broker deals with the establishment and distribution of the encryption keys, the secure storage of the encryption keys.

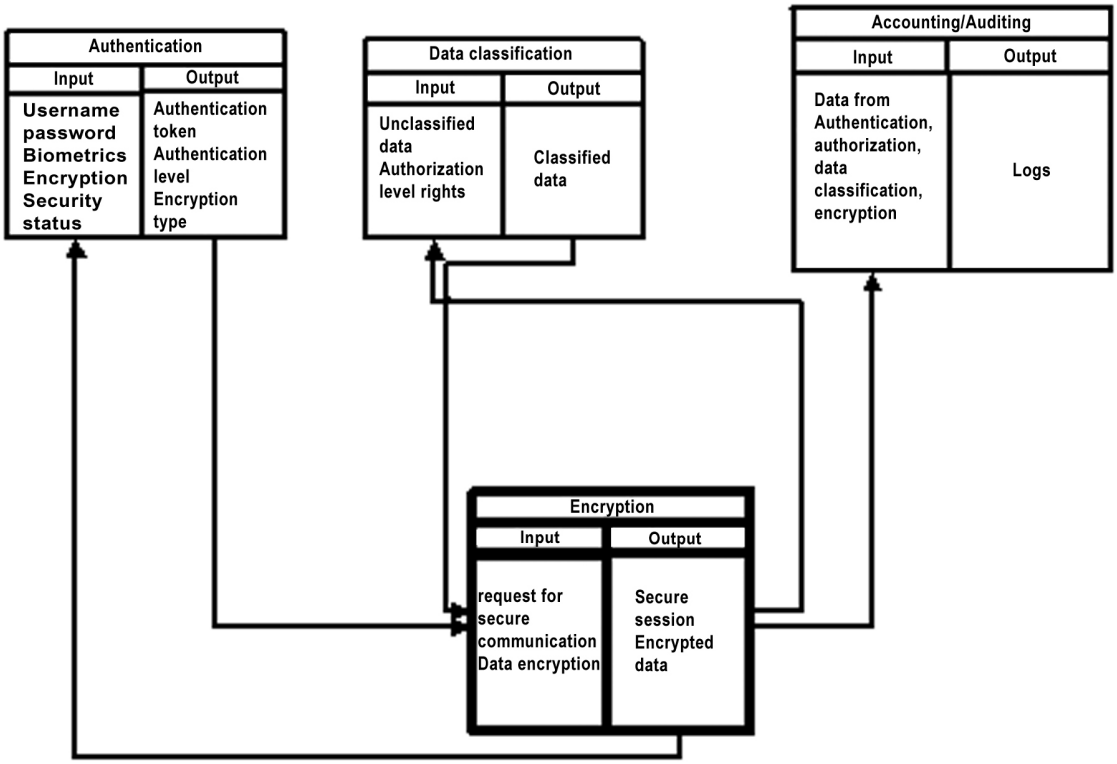


Figure 2 - Encryption process

Trust broker

The main purpose of the Trust broker is to establish and manage trust[5]. It can do this as a local service but can also function as a neutral third party that will facilitate certain services from which it can not take any advantages, except for some compensation from two or more several other parties. If two parties are willing to collaborate, but they know none or little information about each other, they can act through the trust broker. For example if one party provides the service and another one is a customer, the customer may want to determine the trustworthiness of the service and the service can verify if the identity provider performs a real identity check before issuing the identities. After consulting the Trust broker they both can establish a trust of the level, which sufficiently accommodates their needs.

[5] Excerpt from the book "Trust broker Framework" by Adriaan Bruning, Capgemini 2008

Commandments number 6 and 7 refer to the Trust broker:

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”

A Trust broker will act between two or several parties who want to able to do business with each other, but need an extra factor of faith to do this. Trust broker must determine if every party can be trusted and if it is still the same party when the agreement was signed (impersonation case). Therefore the Trust broker creates a circle of trust between two or more several parties.

The Trust broker collects audit data from the domains and through the certain algorithms calculates the risk of dealing with the party. The higher the risk, the lower level of trust should be established.

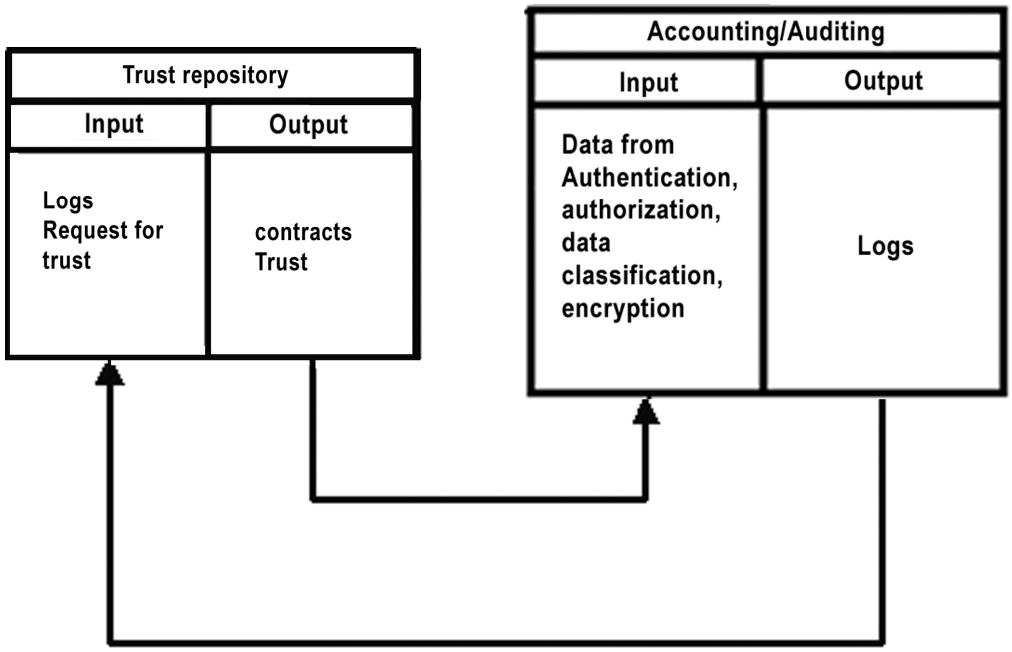


Figure 3 - Trust broker service

Endpoint security

The endpoint security process[6] is responsible for providing the means to establish inherent trust levels between the endpoints. The main purpose of the service is to ensure the device security but also to verify that all the devices involved in a transaction meet the criteria of trust for that transaction. At the moment, many endpoint security and network access control solutions exist. However, most of these solutions were not designed to interoperate with other solutions and lack the ability to verify all network devices. Most solutions provide only endpoint security for personal computers running certain operating systems.

Several of the Jericho commandments refer to the Endpoint Security process:

- The second commandment of Jericho states that "Security mechanisms must be pervasive, simple, scalable & easy to manage"
- The fifth commandment states that "All devices must be capable of maintaining their security policy on an untrusted network"
- The seventh commandment states that "Mutual trust assurance levels must be determinable"

These commandments require a solution where every device connected to a network should be able to participate in the endpoint security process. This means that a universal standard should exist that governs agent behaviour and interactions.

In order to for a secure device to function in a possibly insecure network, devices must be able to maintain their security policies. This means a solution should exist that can monitor device status and act upon it, essentially requiring agents installed on a device.

The endpoint security process is interconnected with several other processes. The authorisation process will be dependent on the endpoint security process to provide authorisation information. In addition, the accounting process will be used to process information gathered by the endpoint security process.

[6] Excerpt from the book "Authorisation and End-point security" by Leon Teheux, Capgemini 2008

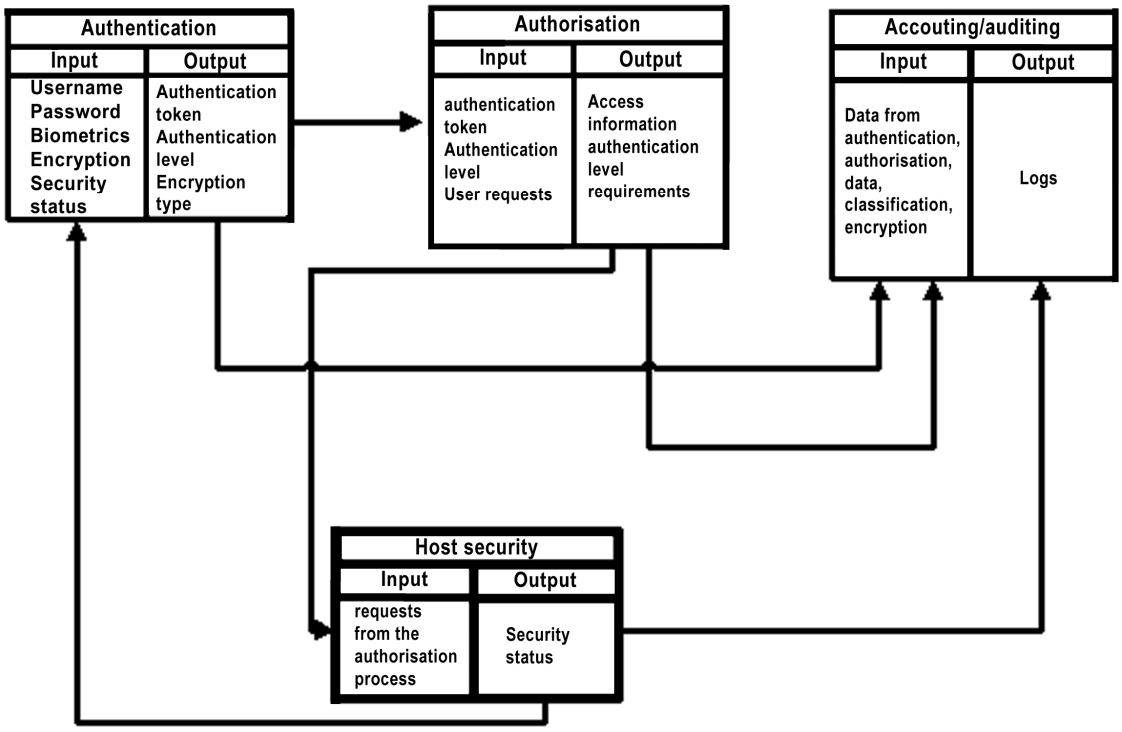


Figure 4 - Endpoint security process

Authentication

According to the 8th commandment "Authentication, authorisation and accountability must interoperate /exchange outside of your locus / area of control", identity data must be not usable only within one domain, but also be inter-exchangeable among multiple parties. Authentication is the process which establishes the subject's identity. Data transaction between entities is only allowed to happen after the subject's successful authentication. Authentication can use a single or multiple factors. The more factors are present the more assurance one receives of the person's identity. The aim of the research is to identify the methods and technologies with which make assertions made in one security domain acceptable in another. Authentication is toughly integrated with other parts of the research. Being an essential part of the AAA framework, authentication delivers identity data, which finds its usage in the authorisation process. Accounting keeps logs of all the authentication transactions for the auditing purposes. Some identity data may also be used by the encryption process to secure either an authentication transaction or data transfer.

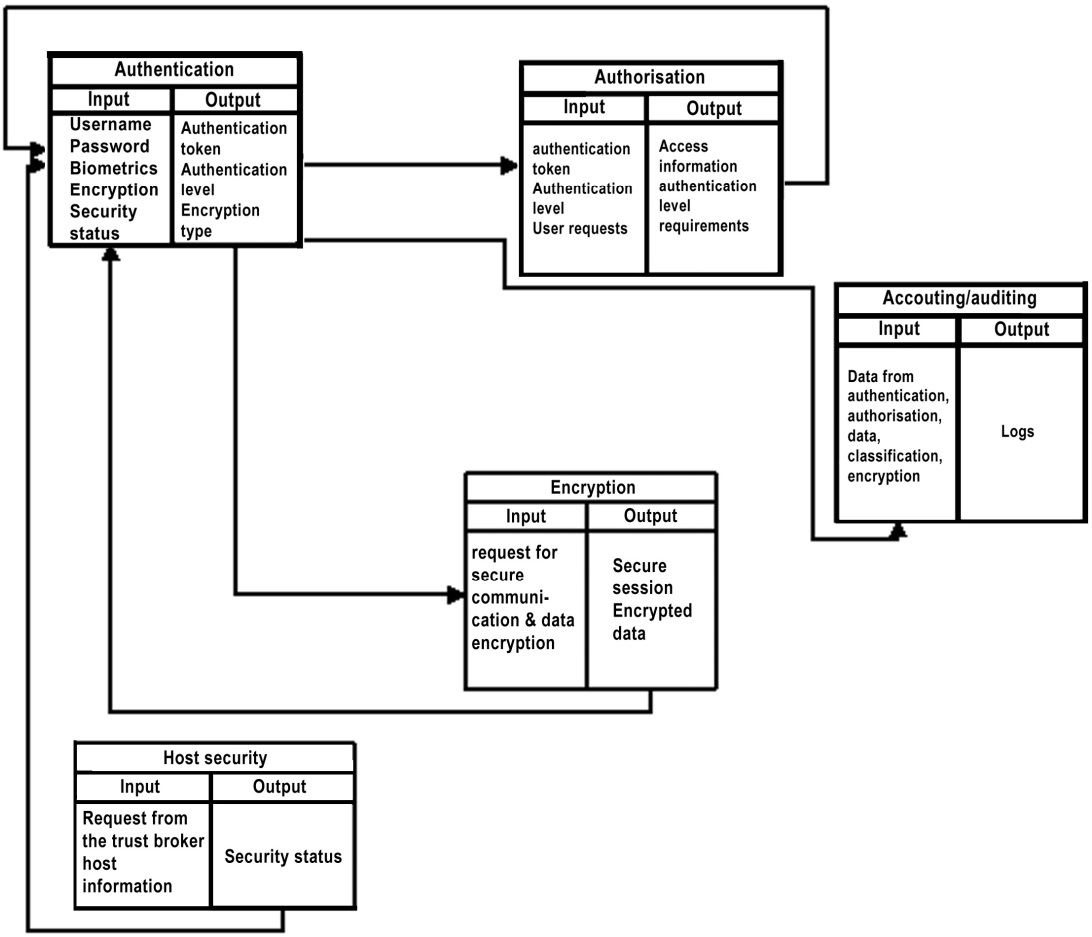


Figure 5 - Authentication process

Authorisation

Authorisation is responsible for determining what rights are applicable to requests. [7] At the moment authorisation is dependent on the authentication process. Only after an entity has established its identity, the authorisation process determines what rights are associated with the resource it attempts to access. Comparing the requirements with the presented data it allows or denies the request. Without authorisation no controlled network interactions can exist.

There are several interactions with other processes. The authentication process needs to authenticate entities before authorisation can take place, whilst the accounting process needs to gather data and process it. In addition, the endpoint security process may need to deliver information that can be used to determine authorisation rights.

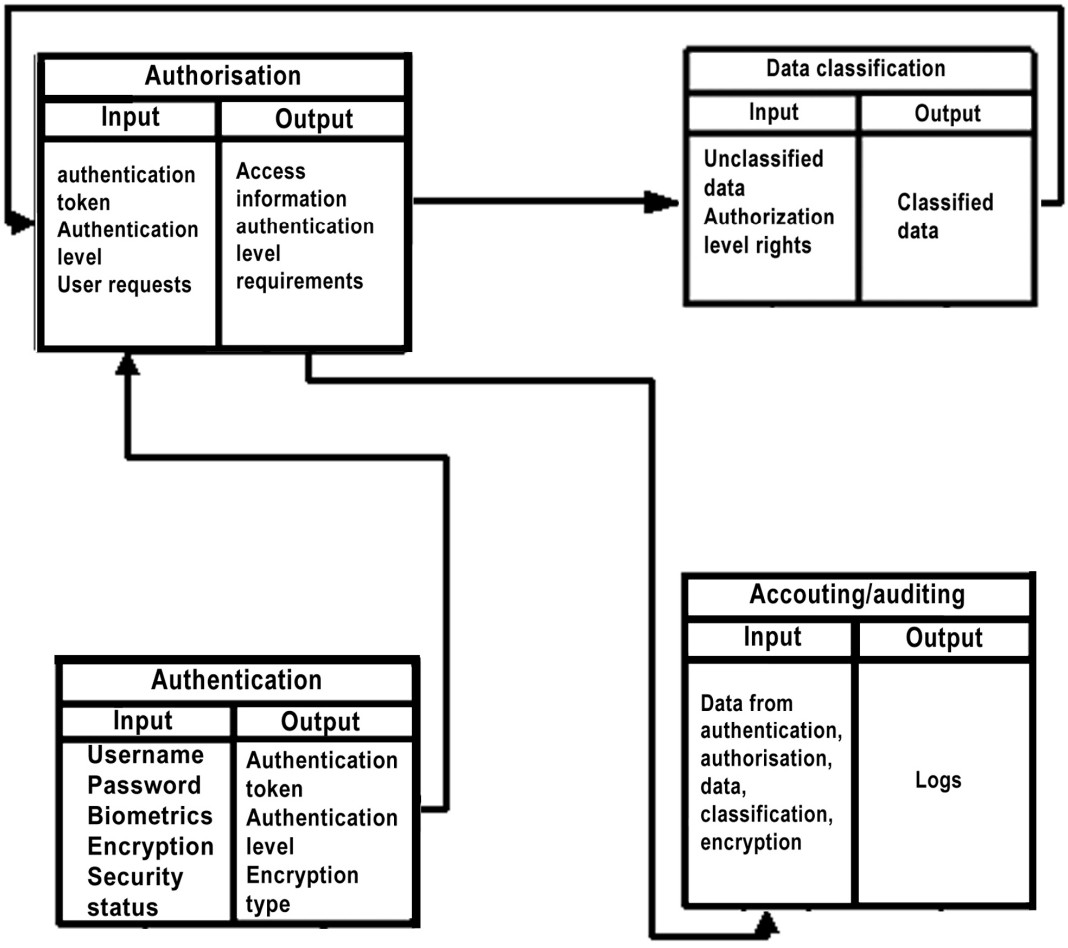


Figure 6 - Authorisation process

[7] Excerpt from the book "authorisation and: End-point security" by Leon Teheux, Capgemini 2008

Accounting

"Authentication, authorisation and accountability must interoperate /exchange outside of your locus / area of control." is stated in the commandment number 8. Auditing is responsible for collection and processing of the log data which is delivered by other processes. Separate IT systems in the enterprise architecture provide log data, which is processed independently from each other. In case of security breach or performance analyses multiple logs must be accessed and analysed. Auditing process may deliver data to other processes, which can determine through the certain algorithms the trustworthiness of the authenticated party. Trust broker service can objectively and independently establish the entity's reputation. Privacy is a very important factor here. No all audit data may be disclosed to the other parties. The goal of this sub-project is to research and define technology, which could consolidate the logs from the multiple systems and provide standardised log data to the analysis processes.

Accounting is connected to every single process. Information from all services and applications is collected and processed in the single log repository.

Data classification

Jericho commandment number 9 "*Access to data should be controlled by security attributes of the data itself*" suggests that data should be able to protect itself. Security attributes are read by the authorisation process and compared to the data delivered by the authentication process. In case of mismatch access is denied. Data attributes are not only used in the authorisation process, but also to achieve the appropriate protection of data and to prevent information leakage. In order to automate the attribute creation process, data should be classified first. At the moment 70% of the enterprise data is unclassified. The aim of the research is to specify the classification process and which attributes can be created.

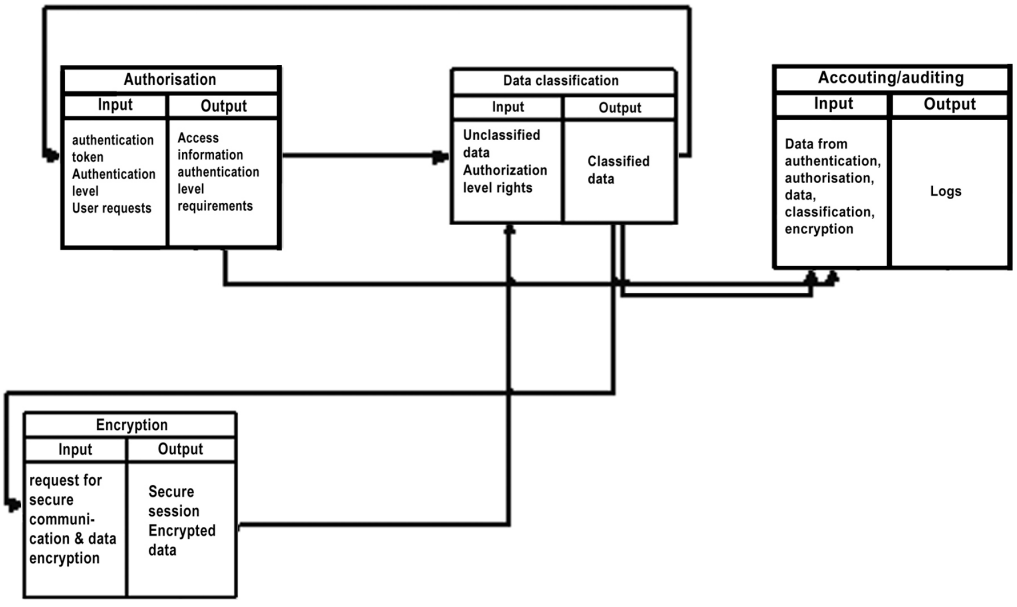


Figure 7 - Data classification process

2. Identity management

"The human experience of identity has two elements:
a sense of belonging and a sense of being separate."
Salvador Minuchin

The collaboration of the modern enterprise with outside world is very often limited by IT resources. Most of the organisations identity management architecture is not ready to extend outside the outer perimeter and create cross-organisational authentication and authorisation schemes. The organisational model has moved from closed groups to an open collaborative environment where interactions between parties define enterprise's vitality. Enterprise needs suppliers, customers and partners. Many business processes are outsourced to other companies which need seamless access to the distributed resources. The evolution of the digital world can be experienced today as we sit at our computers. The development from one host with multiple accounts to global internet collaboration is happening now. Users can't accept anymore that they are locked to a single computer or a single network. Inter-network communication is emerging and the boundaries between the organisations are dissolving. The Jericho forum imposes new requirements upon the identity management. What does an open network mean for the established principles around the digital identities? To answer this question let us first see what identity management is. Enterprise architecture may be composed of multiple systems which interact with each other and which have an isolated directory and security policy. An identity management system spans over the underlying enterprise infrastructure and provides them with a single identity security policy and management standards.

Identity management can be defined as following:
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.8*
Identity Management *is an integrated system of business processes, policies and technologies that enable organizations to facilitate and control their users' access to critical online applications and resources - while protecting confidential personal and business information from unauthorised users.9*

Let us see how interaction between two security domains occurs. If a person from an external domain needs to access the resources in the local domain, then an account will be created which is authorised to access it. This account is managed locally and the external administrator cannot influence it. This means that a person has two distinct identities. One is used in the external domain and another one in the local domain. Every other domain

[8] www.searchvoip.com
[9] en.wikipedia.org/wiki/Identity_management

requires a new account. This creates a complex and inconsistent environment. Here is a summarisation of the weak points of the current identity management principles:

- Redundant work for the network administrators.
- Inefficient and insecure management. Because the domains don't have information exchange, deletion of the account from the primary domain is not reflected in other domains. A person may sill have access to the information he is not authorised to.

What is needed is a simple and consistent environment where a single digital identity that can be managed locally, but which allows a person to access all the resources in the external domains. What challenges faces identity management in a de-perimeterised network? When there is no boundary not only the external identities will be allowed to enter the local security domain and access resources, but the local identities will be accepted in other domains as well. It does not imply that it will broaden the identity management infrastructure to infinity and make it unmanageable. It suggests that every domain will still have their own independent identity management infrastructure and they will exchange information if required.

Digital identity

What defines a person and his or her digital identity? In order to define a digital identity we need to define a person's physiological identity first.

Identity is the set of characteristics that somebody recognizes as belonging uniquely to himself or herself and constituting his or her individual personality for life.[10]

Digital identity that is used in the corporate network or in the Internet must also uniquely identify an entity. Unlike the real world, where only living creatures can possess an identity, a digital identity may also refer to a network service or a network device. Phillip Windley in his book "Digital Identity" defines digital identity as following: "A digital identity contains data that uniquely describes a person or a thing but also contains information about the subject's relationships to other entities". According to this notion a digital identity consists of two parts: data that uniquely describes the subject and additional data.

[10] http://encarta.msn.com/encnet/features/dictionary/DictionaryResults.aspx?refid=1861619974

Digital identity	
Unique information	Additional information: **Attributes** – Acquired data concerning the subject. This data may be shared or personal. This data may also change during the course of time. (Example: Reputation, connections with other subjects, driving license, etc). **Preferences** – Representation of the user desires. (Example: Colour scheme) **Traits** – Inherent data concerning user (DNA, eye/hair colour, date of birth)

Table 1 - Digital identity

Digital identity may be a combinative or separate use of the above standing data. For example a combination of the certain attributes may be enough for an entity to make an identity claim. On the other hand unique information can satisfy the assertion as well. The above standing definition may be reflected on the concept of the Three Tiers of Identity by Andre Durand. [11]

Tier 3: Abstract Identity	Abstract data such as demographics
Tier 2: Shared Identity	Attributes/ Preferences /Traits
Tier 1: Personal Identity	Unique information/ Traits

Table 2 - Three tiers of identity

By elevating from the first to the third tier we acquire more detailed information about the identity. It does not mean that information from the upper two tiers is essential, but it gives additional information which user also might want to disguise. User consent is an important law[12] of identity, which puts the user in charge of which information he is going to disclose and to whom.

Personal Identity is something unique, which is in possession only by a single entity. This identity is timeless and can't be easily modified. It is the true identity.

[11] http://www.digitalidworld.com/modules.php?op=modload&name=News&file=article&sid=26
[12] http://www.identityblog.com/stories/2005/05/13/TheLawsOfIdentity.pdf

Shared Identity may be assigned by others or acquainted in the course of time. For example when a person receives a job, he is assigned to a certain role. When a user registers at the certain online shop, he receives an account with certain attributes.

Abstract Identity says nothing about a person self. It is used for marketing and statistical information. And example of this identity is: female, Dutch nationality, above 40, frequent buyer.

Identity that may cross the boundary of security domain is a complex combination of the unique information, attributes and traits. This information may be static or dynamic. When our real identity changes the digital identity reflects those changes. The most important attribute is the relationship to other entities. When this information is incorporated into the identity it may signify that the bearer of the identity belongs to a certain domain and the requestor should communicate with it in case additional information is needed.

Identity lifecycle

Combined all identity management processes constitute the identity lifecycle. There two questions that must be answered. What distinguishes the identity lifecycle in the open network from the closed one? Which activities may find their place outside the boundaries? All activities concerning the identities correspond to the certain period of the lifecycle:

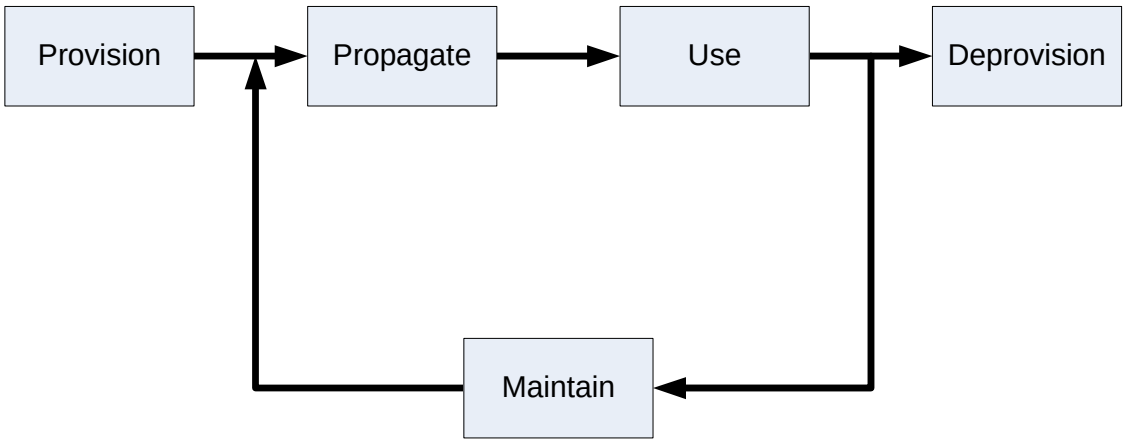


Figure 8 - Digital identity lifecycle

Provisioning

Provisioning is the process of providing of a new digital identity. Provisioning happens through administrator or self-enrolment process. Identity management systems can be straightforwardly connected to HRM system and provision with digital identities as soon as the information about a new employee appears there. Self-enrolment process is used very often at the online shops. They have to deal with large amount of customers and the inefficiency of manual work would be in this case very obvious.

Provisioning doesn't only provide the user account but also creates user attributes which are directly related to him. Every organisation uses its own attribute schema. Some companies are satisfied with only user name and email. Others require all attributes of identity to be filled in. The more attributes are filled in the more systems can use these information.

Propagating
After the record is created it is propagated to other identity resources that are suitable for identity management control. Instead of creating separate user accounts it is created once and through secure protocols reproduced in all systems of the whole infrastructure. Other systems use only attributes that can be used. For example for router and switches the additional accounts information about home address is irrelevant, but for HRM systems this information is vital.

Using
In this stage the user uses his digital identity to gain access to resources. Using is the only process which may extend outside the boundaries of the local domain.

Maintaining
The attributes of a user may change in the course of time. A person may move to another building or receive a promotion. Identity data should reflect these changes and provide up-to-date information to a requestor. Password is the most changeable attribute of the identity. Due to the fact that password reset is the most costly activity of the help desk many identity systems provide self-service.

Deprovisioning
When identity is not needed it is removed from the central repository and consequently from all the systems on the network. Deprovisioning is an important process and can be seen as fraud and crime prevention. After employee leaves the company his account should be blocked / removed as soon as possible. But removing it from the directory service is not a sufficient measure; it should also be removed from all the system where this employee has access to.

Conclusion identity lifecycle
Every identity lifecycle with exception of use limits its activity strictly to the local security domain. Only use may be extended outside the boundaries. Identity has strong bound to its origin domain. Identity may have local usage or be qualified for federation. In no situation it is allowed to be provisioned, maintained or deprovisioned from outside. An external domain may send a request for changes and it will be approved and carried out by the local team. Otherwise it violates the security principles of the identity management.

Conclusion

Identity management faces the challenge of de-perimeterisation. The current principles of the closed network identity management are inefficient and insecure. After the primary identity is deprovisioned other systems may still have accounts related to that identity in use. This puts security of other domains at risk.

Another model is required. An open network implies that identities are passively extended outside the network. A person may use his identity where it is required and they are not bound to the local domains only.

An open network identity is characterised by the complex combination of unique information, attributes and traits. The relationship to other entities is by far the most important attribute. With this one can specify the origination domain of the identity and external domains where it can be accepted.

The use process differentiates the de-perimeterised network from the closed one. Activities of the use process of the identity life cycle may extend outside the perimeter of the network. The other processes are restricted to the local domain.

3. Identity models evolution

From the previous chapters it can be concluded that there is a need for an identity model that can support complex identities and allow its usage outside the security domain. Not every identity model allows this. Let us go through the identity models and define the most suitable one. There are few models and each of them has its own distinct characteristics.

Isolated

Isolated identity is an early form of the identity management. Every single computer on a network has a local user accounts database that is used only by the users working on that computer. The whole network consists of the stand-alone stations. The cost of maintenance of the network increases with every single computer that is added to it. Users from one station can't access data on another station unless he has an account on that computer.

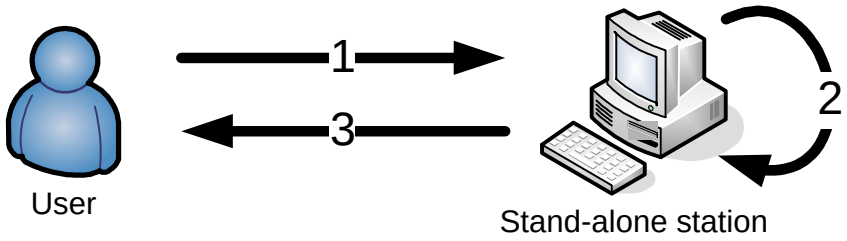


Figure 9 - Isolated identity model process

Process:

1. User logs on to the computer
2. The credentials are checked against the local user database
3. User is authenticated

Data sharing is complicated. If user wants to grant access to a resource on his computer, he needs to create a separate account for that (secure way) or inform another user about his credentials (insecure way).

Identity management

Identity management is very cumbersome. Every user database is managed separately. There is no centralised resource for identity or policy management.

Conclusion

This model is completely unsuitable for Jericho concept, because identity cannot be taken outside the single computer.

Centralised

Development of the identity management went further with the creation of the directory service. A centralized user database is created for identities and policy management. Identities are bound to a single domain. Scalability is very limited. Trusts between domains can extend the scalability to a certain degree. In this case trust serves as an active extension and allows users of both domains the use of the resource. Because of multiple collaboration agreements trust management can become very complicated and cumbersome.

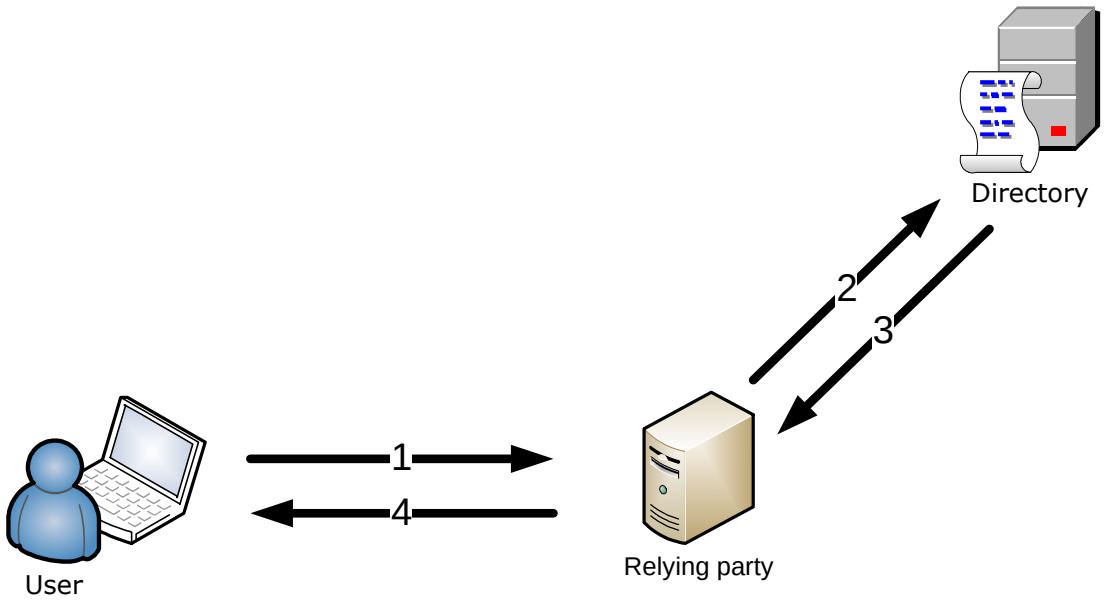


Figure 10 - Centralised identity model process

Process:

1. User request data from the resource.
2. Resource sends an authentication request to the directory server.
3. Directory server sends a reply with information about the directory entry.
4. The user is authenticated.

Lack for support of multiple systems is a weak point of the directory service. Companies with heterogeneous software systems implement a Single Sign-On feature to simplify identity management. Third party software is often needed in order to provide environmental integrity and seamless user experience. The same credentials can then be used on all systems on the network within a single domain. The biggest limitation of the centralised model is the lack of communication between domains. A user is locked to his domain and by default cannot access data outside the perimeter of his network, unless he has an external account for this purpose.

Implementation of the interdomain trust can solve this problem but some unsolved issues may still remain:

- An enterprise does not need to create trust with every company it collaborates with. Interdomain trust creates a deeper trust relationship than it is required.
- Lack of interoperability between different vendors. Domains with different operating systems must be able to agree upon trust standards.
- Two-way agreement is required to create a trust. An enterprise does not need to possess knowledge of the organisations that are going to be consumers of the services and access the resource. They are open for a one-way trust with everyone. (E.g. Google apps)

Identity Management

Users may have multiple accounts in different directories. Network device may be listed only in one directory. Before it can be placed to another directory the whole configuration process has to be repeated again. Identities are stored in a central database. User accounts and policies are managed from a single location. In some situations user is offered an ability to partially manage his identity attributes, which simplifies the job of the network administrator. Collaborating companies have to create databases for internal and external workers. A user is provisioned with an account at every resource domain he needs to access. Upon completion of the tasks or in case job termination the identity management process has to identify all the resources the user has access to and block/delete his account.

Conclusion

In the era of collaboration the limitations of the centralised model become obvious. Resources are dispersed and can be located outside the domain. Interdomain trust tries to solve the problem of collaboration and interconnectivity, but some issues remain: trust relationship which is established is deeper than required, mutual agreement is needed and lack of interoperability between different vendors.

The centralised model qualifies for a de-perimeterised concept up to the certain degree. With the help of trust one can achieve communication with other domains but when extensive collaboration is required it fails to fulfil its mission.

Federated

To overcome the collaboration and scalability limits of the centralised model enterprises move to the identity federation or Identity 1.5 as Dick Hardt[13] named it. Federation allows for a digital identity to cross the perimeter and to be accepted by another security domain without cumbersome interdomain trust schemes. Federated identity model was recently developed and at the moment there are few emerging technologies that make the implementation possible. Burton Group defines Identity Federation as follows: “The

[13] www.sxip.com

agreements, standards, and technologies that make identity and entitlements portable across autonomous domains". BeID[14] is good example of the Identity Federation. Every single resident of Belgium receives an electronic ID card that can be used to sign onto different government websites or encrypt and put an electronic signature on the emails. The Belgium government serves as an identity provider and a trusted authority to which one can refer in order to verify the user's assertion.

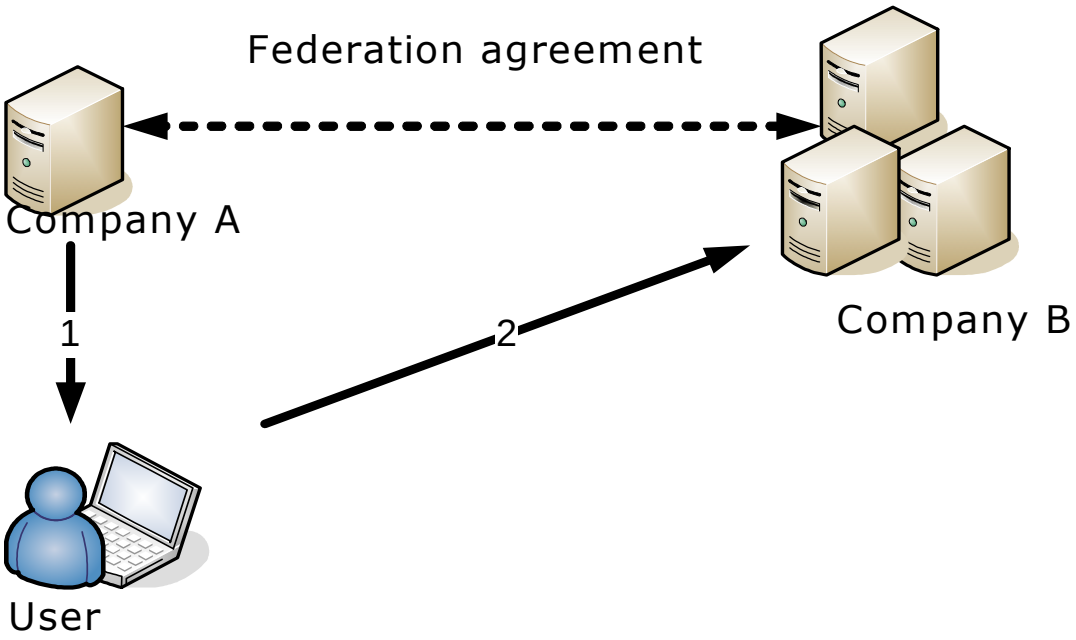


Figure 11- Federated identity model process

Process:
Company A establishes a federation agreement with Company B.

1. The user receives his digital identity from Company A (Identity Provider).
2. A digital identity is presented to the Company B (Relying Party) in order to access their resources.
3. (optional) Relying party may verify the identity from the Identity Provider.

The authentication methods and identity flow will be discussed in greater detail in the authentication chapter. Every identity system has its own methods and ways in which the identity data can be delivered to relying parties. Further the identity systems will be closely compared and I will give an advice about each of them. There are three possible topologies of identity federation[15]. They also outline the evolution of the enterprise's collaboration with other companies.

[14] http://eid.belgium.be/
[15] Windley, Phillip. Digital Identity. O'Reilly 2005.

Ad-hoc federation

Two enterprises agree upon unilateral or bilateral federation of the digital identities. E.g. Company may provide access to its resources when some tasks are being outsourced.

Hub-and-spoke federation

A federation island formed around a big company. Unilateral or bilateral federation agreement is signed by two or more companies. E.g. Google provides access to its online application and allows users to use single sign-on feature. Identity data is transferred from a company domain in a SAML token and accepted by Google. The absence of the common identity system may hamper the acceptance of this pattern. Unilateral federation agreement is possible in two scenarios: service provider and identity provider. [16]

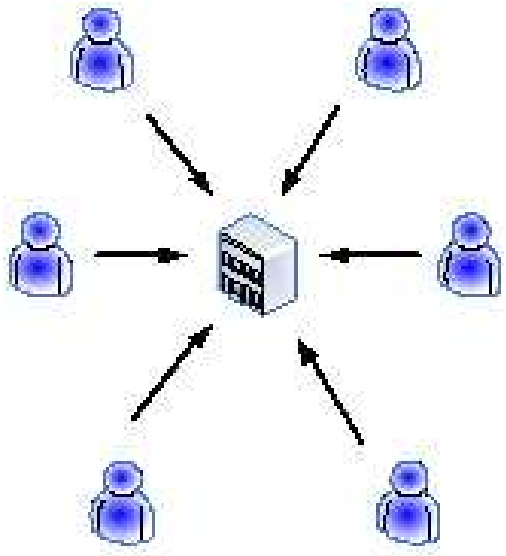


Figure 12 - Service provider topology

Service provider

This infrastructure is characterised by a single relying party and multiple identity providers. The parties have agreed upon the unilateral identity acceptance by the service provider. Service providers have a certain level of trust at which entities are identified before they are provisioned with a digital identity. Strict security policy may dictate that an entity should provide a valid ID card for a validation. Less strict policy would be satisfied with a self-enrolment process, when a user provisions himself with the digital identity.

Identity provider is characterised by a single identity provider and multiple relying parties. A federation agreement describing unilateral acceptance of the identities from the single

[16] White Paper. Federated Identity Primer by Ping Identity. http://www.pingidentity.com/download/show/289

provider is signed. Identity systems may vary per service provider. An example of such an identity provider may be government which provisions its citizens with electronic ID cards and provides access to the scattered government resources.

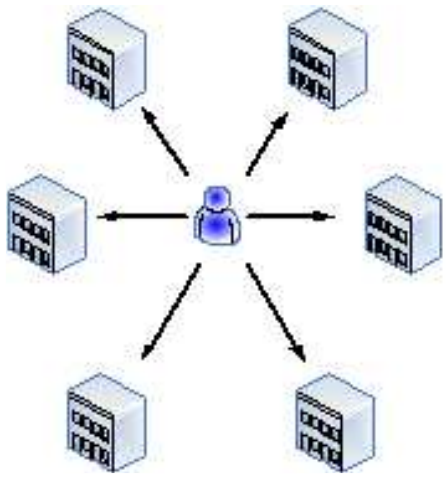


Figure 13 - Identity provider topology

A bilateral federation agreement among multiple providers would create the third scenario – **Multi-Provider Cross Domain.** Parties are functioning as the service provider and identity provider simultaneously. Management of the agreements and authorisation of the external identities is becoming a challenging task. The more relationship one party has with other the more complicated the task of management becomes. The complexity of the network increases with every single node added to the federation.

Identity federation network

Identity federation network is an organised identity environment where organisations and individuals can freely interact and collaborate. It is privately owned or managed by a non-profit organisation. Common open standards, security policy and privacy are defined and enforced through all network members.

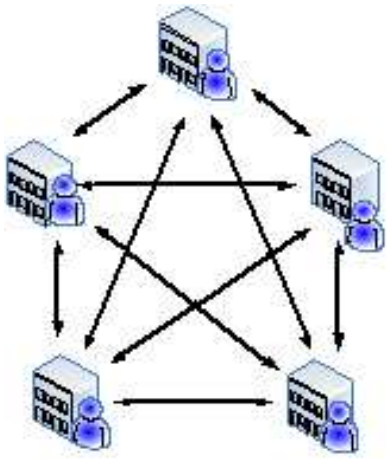


Figure 14 - Multi-provider cross domain topology

Identity management

Federated identity management consists of two parts: account management and federation agreement management.

Local accounts are managed within the local domain boundaries and are not reachable from outside. Local management doesn't differ much from the centralised model. User accounts are stored in the centralised database.

Federation agreement management depends upon the agreement type. Unilateral agreement implies that one side provides services and information. The provider passively agrees upon the federation. In other words it stays open for the collaboration. Bilateral agreement implies that both sides actively participate in the collaboration. On your side you define whether you provision users with identities that they can take outside the enterprise perimeter and whether you authorise identities from other enterprises to access the resources inside your management domain.

It is important to note that a company may theoretically have an infinite number of federation agreements. So besides the management of the accounts on the network administrators will have to administer the federation agreements between companies. Trust broker handles these agreements and also authenticates the external users. The more systems can be reached with the trust the more complex the whole infrastructure becomes.

Conclusion

The federation model allows identity data to be exchanged between the domains. The main advantage of the federation model is that it can be build using current infrastructure. Domain boundaries stay intact and identities are still managed by the administrators. The only difference with the centralised model is that it allows collaboration in simplified, but still secure way. Federation is not a trust but a passive extension of the identity data outside the boundary.

User centric

User centric model is actively promoted by Dick Hardt from Sxip Identity. His concept of Identity 2.0 (the next step after federation) places user at the centre of all interactions between an identity provider and the relying party. This model is the true translation of the real world where every person is in charge of his identities. A very simple comparison would be a wallet where a person keeps his bank cards, customer cards and ID's. He knows where to show which card and whether he desires to do that. When a card is stolen it can be revoked just as simple as one revokes his credit card. In this case a person receives a new card or it can be also said a new digital identity.

User centric model works with claims. A person claims that he is what he says he is. In other words I claim to be me. Sounds ridiculous but if we look back at the real world this is what we do. Hardly ever someone doubts in the truth of your words and even if they do you can always prove your identity by showing one of the ID cards you have in your wallet. In order to prove you assertion in the digital world you need to authenticate yourself. Unlike other models, in the user centric model authentication can happen also on the computer of the user. This requires a secure agent which handles the authentication process. After the user authenticates himself his claim reaches the relying party where a decision is made whether to trust this claim or not.

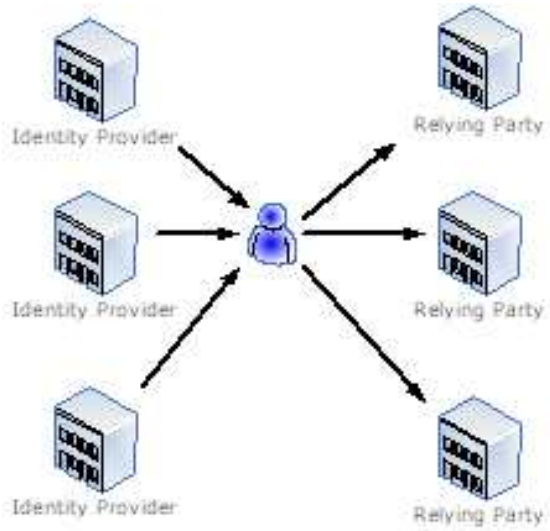


Figure 15 - User centric identity model

On the left side of the picture you can see Identity Providers (IP). They verify the claim assertions by authenticating user or underlying claims (claims can be based on other claims). After the claim is verified, the identity provider issues a security token which is presented to the Relying Parties (RP). The authentication process and identity flow will be covered in more detail in the Authentication chapter.

User consent is a very important factor in user centric design. Real world authorities do not know where the ID card is presented by a person. This creates a certain level of privacy which is absent in the virtual world. In a user centric design the identity provider does not keep track of user actions because the only requests it receives are from the user agent. It serves as a medium between the authority and the service. 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 are few trust models scenario's which are discussed in the chapter "Authentication and Trust". Trust management is covered in more detail by Adrian Bruning in his thesis "Jericho Forum Project: Trust broker".

Identity management
Identity management takes a completely new path here – it places user in charge of his identities. So what is managed where in this model?
Identity provider:

- User identification
- Identity provisioning
- Identity maintenance
- User authentication
- Identity revocation
- Trust management

User:

- Identity provisioning (self issued)
- Identity maintenance
- Identity revocation
- User authentication

Relying Party:

- Trusted identity issuers management
- Security token verification
- Security policy
- Authorisation policy

We can see the clearly defined separation of the user, identity provider and data resources. Each of them has its own management domain. Unlike the previous models users receive responsibility of administration of their identities and consent of the information they send to the relying parties.

User centric model truly erases boundaries as envisioned by the Jericho Forum. Users do not belong to any domain any more. They have an identity issued by that domain and they choose when and where they present this identity. The user has strong consent of which information is transmitted to the relying party. Identity usage is in the hands of the user. We can say that the user centric model is the translation of the everyday principles of the real life into the digital world. 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. This will ensure a smooth and seamless further development. The reason for that is that a company will have already worked out schemes of assertion issuance and federation agreement management. It is also possible to implement the user-centric model for usage with the relying parties first and then gradually dissolves the local domain boundary. But the user may experience some problems when the identity management will be fully shifted to them. After all they are not used to it yet.

Conclusion

The most important requirement of the Jericho forum is to allow a user identity to extend outside the security border. There are three models which allow this: centralised, federated and user centric. The centralised model does not allow it per default and takes advantage of interdomain trusts. There are certain disadvantages of this model. Trust management is cumbersome and it allows establishing only deep relationship between companies. Domains from multiple vendors do not interoperate with each other. Finally it requires mutual agreement of both parties to establish a trust. Because of these disadvantages the centralised model won't be considered for a de-perimeterised network. The federation model solves the issues of the centralised model and allows exchange of user data without changing the current infrastructure. Federation agreement management is the process that will add the complexity to identity management. The user centric model is the true solution, which is envisioned by the Jericho Forum for a boundaryless Internet. It dissolves the borders of the domains and allows free and secure communication. This model requires revision of the current infrastructure and it is advised to choose the gradual approach. Both of these models allow companies to exchange the user identity data and bring collaboration to the next level. Unlike the federated model, the user centric model puts user in charge of his identities. It also eliminates complicated federation topologies of the trusted enterprises, which reside in the privileged circle of identity data exchange. In the user centric model the user defines his own trust perimeter. With all its advantages the user centric model still partially remains in concept and needs further standardisation.

The way we work with information has evolved from interaction with a single computer with a shared disk space to the global information network – Internet. Resources are dispersed all over the network and the local account database cannot extend that far. Evolving from the isolated model to the federation model has taken a long time. It will also take a while to implement user centric model. But in my opinion this is the only way of the virtual identity evolution.

4. Identity systems

We have concluded that federation and user-centric models are the most suitable for a de-perimeterised network. Now there is need to determine which identity system is the most acceptable and which would meet the needed requirements. *Identity system* is a complex collection of protocols and elements that work together and allow identity management from inside and outside the security domain. From the several identity systems there are few which became widely accepted and that are being implemented in the identity management products. Identity systems support more or less features of the federated or user centric models. The analysis of the identity systems is performed from an enterprise user prospective. This means that the following research topics are essential: user authentication methods, identity flow and identity federation.

What are the logical requirements for these systems? The best requirements until now were published by Kim Cameron in his white paper "The Laws of Identity"[17].

1. *User control and consent*
   The information is allowed to be disclosed only with the user's consent.
2. *Minimal disclosure for the constrained use*
   Identity system should reveal the least amount of information possible.
3. *Justifiable parties*
   Only the required parties are allowed to receive the identifying information.
4. *Directed identity*
   Omni-directional identifiers and unidirectional identifiers should be supported.
5. *Pluralism of operators and technologies*
   No monopoly of the identity providers should exist.
6. *Human integration*
   The human must be a component of the distributed identity system.
7. *Consistent experience across contexts*
   Separate context should be experienced by the user consistently.

Until now there is no single identity system that can satisfy the above standing requirements. Microsoft is developing its own identity metasystem, which is called Microsoft CardSpace™ (formely Infocards). It is important to note that Kim Cameron is an employee at Microsoft which means that it is highly possible that the promoted identity metasystem should meet these requirements. Until then they are used as the reference point of the ideal identity system.
Below there is a description of several identity systems. They are placed into two logical groups: federated and user-centric identity systems. Though we came to the conclusion

[17] http://www.identityblog.com/stories/2005/05/13/TheLawsOfIdentity.pdf

that user-centric system needs standardisation, there are multiple serious initiatives which have to be considered for the future deployment.

Federated identity systems

SAML

SAML is the product of Organisation for the Advancement of Structured Information Standards OASIS[18]. SAML stands for Security Assertions Markup Language. It is an XML framework to exchange authentication and identity attributes between the security domains. There are several versions of SAML.

- SAML 1.0 was adopted as an OASIS Standard in November 2002
- SAML 1.1 was ratified as an OASIS Standard in September 2003
- SAML 2.0 became an OASIS Standard in March 2005

SAML v2.0 is incompatible with its predecessors and to be more specific the assertions and protocol messages are different. Technical solutions may use different versions of SAML which may cause interoperability problems. Before implementing it one should make sure that it uses the compatible version of SAML.

SAML consists of several components: assertions, protocols, bindings and profiles.[19]

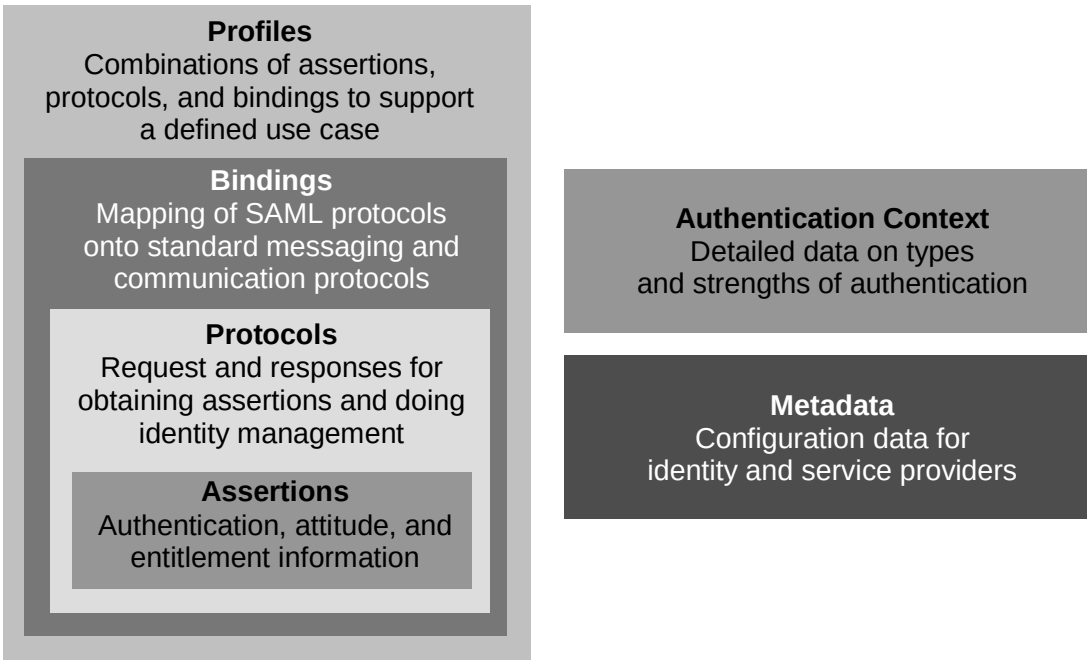


Figure 16 - SAML 2.0 architecture

[18] http://www.oasis-open.org

[19] http://www.oasis-open.org/committees/download.php/13525/sstc-saml-exec-overview-2.0-cd-01-2col.pdf

Assertions: An assertion is a package of information that supplies one or more statements made by a SAML authority. SAML recognizes three types of assertions: authentication, attribute and authorisation decision.

- *Authentication assertion* states that a specified subject was authenticated by particular means at the particular time.
- *Attribute* specifies additional information associated with the subject.
- *Authorisation decision* specifies whether access to a certain resource is granted or denied.

Protocols: SAML has a number of defined request/response protocols.

- *Authentication request* protocol allows making request from the SAML authority about authentication and certain attribute assertions.
- *Single logout protocol* defines mechanisms by which all active sessions can be near-simultaneously terminated.
- *Assertion query and request* protocol defines a set of queries by which certain SAML assertions may be acquired.
- *Artifact resolution protocol* provides the means by which the SAML protocol messages may be passed by reference using a small, fixed-length value called the artifact.
- *Name identifier management protocol* provides mechanisms to change the value or format of the name identifier used to refer to a principal.
- *Name identifier mapping protocol* provides a mechanism to map one SAML name identifier into another.

Bindings: SAML bindings define how SAML protocol messages can be carried in underlying transport protocols.

- *HTTP redirect binding* defines how SAML messages can be transported using HTTP redirection messages.
- *HTTP post binding* defines how SAML protocol messages can be transported within the base64-encoded content of an HTML form control.
- *HTTP artifact binding* describes how an artefact may be transported using HTTP protocol.
- *SAML SOAP binding* describes how SAML messages may be transported within SOAP messages.
- *Reverse SOAP binding* defines a multi-stage SOAP/HTTP message exchange that permits an HTTP client to be a SOAP responder.

- *SAML URI binding* defines mechanisms for retrieving the existing SAML assertion by resolving Unified Resource Identifier.

Profiles define how SAML assertions, protocols and bindings can be combined in order to achieve a greater interoperability in certain usage scenarios. SAML 2.0 defines the following profiles: Web browser SSO, Enhanced client and proxy, Identity provider discovery, Single logout, Assertion query/request profile, Artifact resolution profile, Name identifier management and name identifier mapping. For more information about profiles, their usage and other technical details concerning SAML 2.0 I refer to the SAML technical overview[20]. This document provides comprehensive information about SAML 2.0 and the differences with the previous versions.

SAML is a complex system and a standard which 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

Liberty Alliance was formed in 2001 by 30 members to establish open standards and best practises for identity federation. Since its establishment the consortium has grown up to nearly 150 organisations. The Liberty Alliance works openly with other organisations adopting the current published standards and contributing relevant work.

There are three important specifications published by the consortium:

- **Identity Federation Framework (ID-FF)** [21] consists of the core specifications that make the creation of the multivendor identity federation network possible. Liberty Alliance realized that a convergence of the standards would further promote the adoption of identity federation. Their set of specifications was contributed to OASIS for the development of SAML 2.0. These specifications enable identity federation and management through features such as identity/account linkage, simplified sign on, and simple session management.

- **Identity Web Services Framework (ID-WSF)** [22] is a general framework for discovery and invocation of identity services. After the user has authenticated itself at the identity provider, his assertion can potentially be used by the relying party to discover services this user is eligible for. These specifications provide the framework for building interoperable identity services, permission based attribute

[20] http://www.oasis-open.org/committees/download.php/20645/sstc-saml-tech-overview-2%200-draft-10.pdf
[21] http://www.projectliberty.org/index.php/liberty/resource_center/tutorials/liberty_quick_start
[22] http://www.projectliberty.org/liberty/content/download/1297/8256/file/liberty-idwsf-disco-svc-v1.2.pdf

sharing, identity service description and discovery, and the associated security profiles.

- **Identity Services Interface Specifications (ID-SIS)** [23] describes how a service that supports identity information of a principal self should function. This service provides user and user attributes information. These specifications enable interoperable identity services such as personal identity profile service, alert service, calendar service, wallet service, contacts service, geo-location service, presence service and so on.

These specifications may be used together or separately.
The Liberty Alliance provides standards by adopting other standards and delivering own proposals. Compliance with the above described 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
WS-Federation is intentionally described apart from the rest of WS-* languages. It is not included in the MS Identity Metasystem, but it can be used together with other WS-* protocols to federate a user identity. SOAP clients and web services can use the features of WS-Federation directly. Just as the Identity Metasystem it operates with the variety of the security token services and it is heavily dependable from WS-Trust and WS-Security Policy. WS-Federation includes mechanisms for brokering of identity, attribute discovery and retrieval, authentication and authorisation claims between organisations and protecting of the privacy claims between organisational boundaries. WS-Security defines mechanisms to assure message authenticity, integrity and confidentiality through the use of the security tokens.[24]

WS-Federation exists for a long time, although it hasn't found wide acceptance among organisations. Critics say that the reason is the fact that it is heavily dependable on other WS- protocols. Though WS-Federation offers similar possibilities as SAML 2.0, it still misses some important features like for example broad authentication context.* [25]

Comparison of WS-Federation and SAML 2.0
Hubert A. Le Van Gong in his blog[26] gives a comprehensive comparison of WS-Federation and SAML 2.0. Bold denotes advantage.

[23] http://www.projectliberty.org/liberty/content/download/1028/7146/file/liberty-idsis-pp-v1.1.pdf
[24] http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-fed/WS-FederationSpec05282007.pdf
[25] http://blogs.sun.com/hubertsblog/date/20070302
[26] http://blogs.sun.com/hubertsblog/date/20070302

Topic	WS-Federation	SAML 2.0
Target	• Browser Redirect (messages in URL) • Browser POST (messages in HTML form) • SOAP (over HTTP) • Artifact	• Browser Redirect (messages in URL) • Browser POST (messages in HTML form) • Artifact (reference to assertion + SOAP call) • SOAP (over HTTP) • Reverse SOAP (over HTTP)
Security tokens supported	• Those supported by WS-Trust (SAML assertions, X509 certificates, kerberos...)	• SAML assertions • Any other token types (embedded in a SAML assertions via the SubjectConfirmation element)
Dependencies	• WS-Trust [1], WS-Policy, WS-SecurityPolicy. • WS-Eventing to subscribe to Single Sign Out messages. • WS-Transfer & WS-ResourceTransfer.	None!
Identity federation	• Performed by the Pseudonym service (optional...) which provides identity mapping and its management. • A Pseudonym service may be independent of an IP/STS and could store tokens associated to a pseudonym.	• Identity mapping is part of the IdP. Although less (?) flexible it avoids the need for yet another protocol between the pseudonym service and the assertion generator (IP/STS in WS-*).
	• Mapping can be created either by the requestor (principal...) or the owner of the resource (SP).	• Mapping is created by the IdP but can be changed by either the IdP or an SP.
	• All operations on pseudonyms (get, set, create or delete) are done via WS-Transfer (and its extension WS-ResourceTransfer to filter the scope of these operations).	

	• Client-based pseudonyms: a requestor can specify (in an RST) ad-hoc data for a pseudonym it wants to be used by the STS (e.g. PPID, DisplayName, email...)	• SAML does not provide a similar concept to the ClientPseudonym in its AuthNRequest. Is this one of the active requestor "benefit"? The Name ID management protocol (and SPProviderID) is not meant for transient ID mapping.
Metadata	• Description of the federation metadata format. • Description of a secure transfer of this metadata. • Can hold info about several federations.	• Description of metadata for SSO and more. • Organized by roles (IdP, SP, Attribute requester, PDP...)
Single Logout	• Can be initiated by either an SP or the (primary) STS which will send sign-out messages to all RP.	• Similar
Artifact	• Based on the use of a reference token (i.e. an EPR to which a WS-Transfer GET can be made to retrieve the actual token).	• Artifact profile (complete SAML response) • URI binding (to only obtain SAML assertion) • **SAML also defines mechanisms to request or query existing assertions (by subject or statement type).**
Authorisation service	• Again a specialized STS. • Concept of authorisation context (name-scope-value) to condition the issuance of a token.	• The context seems to be a kind of pendant to the SAML2 XACML profile...
Authentication freshness	• A requestor can specify its freshness requirements (allow caching of security tokens etc.)	• Similar with Conditions and ForceAuthN
Authentication level	• WS-Trust defines the parameter (AuthenticationType). WS-Fed specifies predefined values (e.g. Ssl, SslSndKey, smartcard).	• SAML 2.0 offers a much broader & extensible set of authentication contexts**.**
Privacy	• A requestor can express its protection requirements for security tokens it requests (protectData w/h claims & confirmation from STS). • Privacy statements can be retrieved via WS-Transfer.	• SAML offers a range of options to constraint the use & scope of an assertion (audience, advice, proxyRestriction, oneTimeUse, condition) [2] • Those constraints can originate from both the SP or the IdP.

Table 3 – Comparison of WS-Federation and SAML 2.0

Adopted from http://blogs.sun.com/hubertsblog/date/20070302

Both of these protocols allow building federated identity network. SAML 2.0 offers an all-in-one solution with extensibility that is offered by Liberty Alliance standards. It has no additional dependencies and it has a richer authentication context. WS-Federation has an advantage by providing pseudonyms to the users but this is not essential feature for a de-perimeterised network.

Both SAML 2.0 and WS-Federation offer similar features. But SAML 2.0 offers a broader authentication context that is one of the primary requirements for a secure network. It also has no dependencies from other protocols thats puts it ahead of WS-Federation. Dick Hardt has also noted that he doubts that companies like Yahoo or Google would implement a protocol that was developed by Microsoft[27].

User centric identity systems

User centric systems haven't found broad acceptance yet. They offer users much more advantages and simplicity than federation systems but they are still very immature and need proper testing. Until now there are only a few user centric identity systems which can be found on the market: Open ID and Microsoft Identity Metasystem.

OpenID

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. This method is more suitable for a global internet-scale identity system than an enterprise. Organisation requires 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. Sun Microsystems has recently announced that they will provide all its employees with OpenID identities.[28] OpenID may be used in combination with other identity systems such Microsoft Infocards, which would provide more security and additional information that OpenID would not be able to deliver without user agent (e.g. authentication strength, multiple claims, security information).

The advantages of Open ID are its simplicity and lightweight trust model. The biggest disadvantage is the security. OpenID is an open source protocol and recently multiple security flaws were discovered.

[28] http://biz.yahoo.com/prnews/070507/sfm045.html?.v=92

WS-* and Microsoft Identity Metasystem

Windows CardSpace (formely Infocards) is a piece of client software that enables users to provide their digital identity to online services in a simple, secure and trusted way. [29] User software (CardSpace) can be compared to a wallet where user keeps all his identity cards and presents them when needed. CardSpace provides additional security to the user shielding him from phishing attacks by authenticating the relying parties.
Microsoft Identity Metasystem architecture is claimed to use open standards and incorporate multiple protocols that make interoperability between multiple standards possible.

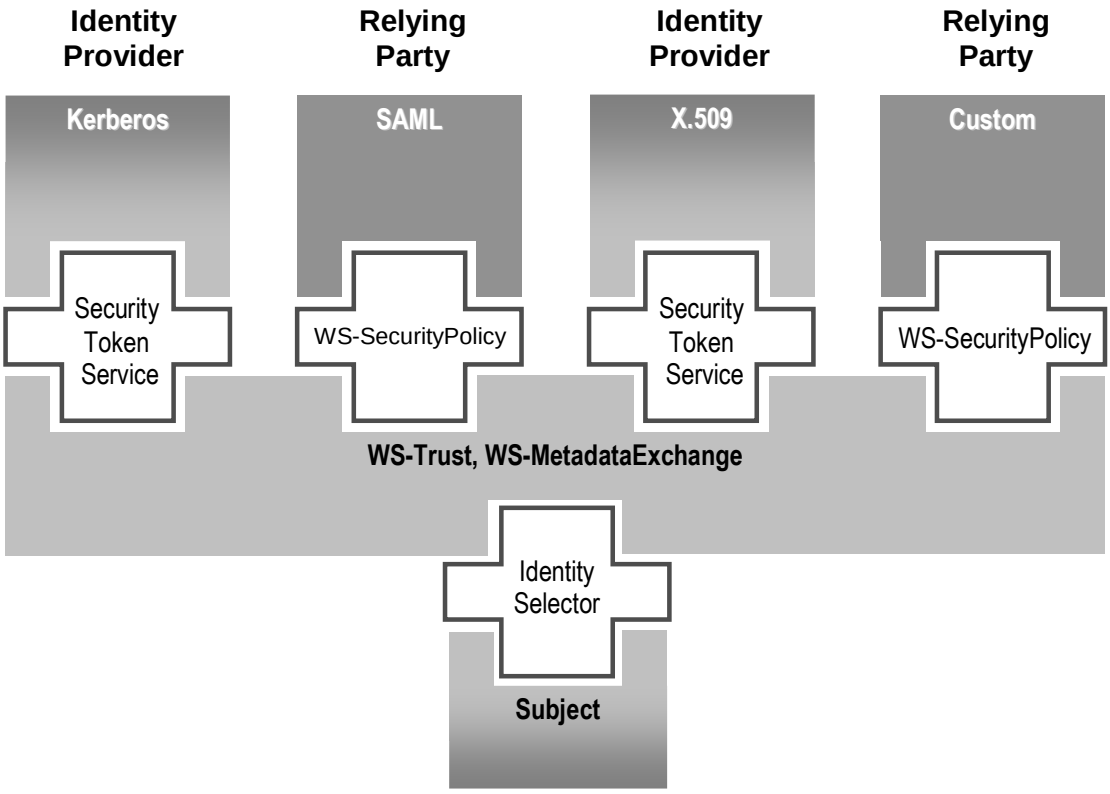


Figure 17 - Windows CardSpace and Identity Metasystem

The components of the identity metasystem architecture are the following:

- The user agent is Microsoft CardSpace™. There are few open source initiatives which provide OS agnostic user experience. Some agents also support multiple factor authentication.

[29] http://cardspace.netfx3.com/content/introduction.aspx

- Identity Provider or Security token services are the identity providers which supply user with and authentication token. Kerberos and X.509 security tokens are supported.
- A relying party is a web service or an application which sets the requirements for identity and claim assertions.
- Languages which make the conversation between user agent, identity provider and the relying party possible. The languages which are the heart of the identity metasystem are called Web Services-* (WS-*). Microsoft and IBM have together developed a set of WS-* protocols. IBM describes the WS-* as following: "*Web services are a loosely-coupled, language-neutral, platform-independent way of linking applications within organizations, across enterprises, and across the Internet. A key benefit of the emerging Web services architecture is the ability to deliver integrated, interoperable solutions -- which makes it critical to ensure the integrity, confidentiality, and overall security of these services*."[30]

Web services may be used together or independently. Every one of them solves a particular problem in web services interoperability. Web services are the foundation of the identity metasystem. They allow interoperability between different identity providers and relying parties. No matter what the security token from the user is, it can be transformed into a token which is required by the relying party.

- **WS-Policy** is a language that describes the security policy of the certain web service: SOAP message security, WS-Trust or WS-SecureConversation. Through this framework a web service may express its security policy and declare how messages are to be secured.
- **WS-Trust** is a language which allows one security token to be exchanged for another. The specifications define dissimilation and issuance of the security tokens within different security domains.
- **WS-MetadataExchange** is a language that defines how metadata associated with a web service endpoint may be represented as a resource and how this metadata may be retrieved from the web service endpoint.

Microsoft Identity Metasystem is complex system which 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.[31]
Identity Metasystem and CardSpace software are still in beta testing, also the open source user agents are in development. But the advantages of the system are obvious. The scope of use can be limited to an enterprise but it can also be used globally on the Internet (as initially intended). Interoperability among multiple standards is achieved and it allows for a security token acquired at one identity provider to be transformed and presented in the

[30] http://www.ibm.com/developerworks/webservices/library/specification/ws-secpol/
[31] http://www.linux-mag.com/id/3353/

needed format to a relying party. There is no centralized or federated identity management. The user is fully in charge of his identities. This allows me to say that identity metasystem is suitable for the user centric model.

Conclusion

SAML has become a de facto standard in identity federation, but it is complicated system which can in the beginning seem to be very confusing. There are also several versions which are incompatible with each other. When choosing the product one should pay attention to the supported versions. The Liberty Alliance incorporates the other standards on the market and delivers proposals for a single standard. By advising and delivering own specifications and by publishing frameworks and requirements they promote certain standardisation. Compliance with Liberty Alliance would give you certain confidence in interoperability, but they don't deliver any identity systems on their own. WS-Federation delivers similar features as SAML 2.0, but closer comparison identifies some advantages of the latter. Also it is doubtful that Microsoft rivals Yahoo and Google would adopt this standard. OpenID is a lightweight and simple protocol which enables internet scale identity, but the current version has serious security flaws.[32] The second version of the protocol is promised to be more secure and feature rich.

In order to provide employees with OpenID identities a company has to create a homepage for every single worker. Which some enterprise may find unacceptable. Microsoft is developing an identity metasystem which would enable great interoperability among several standards and put users in charge of their identity. But this metasystem is at the moment very immature and no non-Windows compatible user agents are developed yet (vendor independency is a primary requirement of Jericho Forum). The Microsoft identity metasystem enables the user centric model but their identity system is at the moment very immature and requires some time to proof itself in order to be implemented within organisations. After receiving support from the Liberty Alliance SAML became a real market leader in the identity federation.

[32] http://kronkltd.net/blog/?p=764

5. Authentication

Security systems are created to allow access to authorised people and keep unauthorized people outside. Three steps are involved in this process: identification, authentication and authorisation.[33]

- Identification. Person identifies himself with a token or a secure string.
- Authenticaion. After the token is accepted, a person has to provide evidence of his identity. At this step the identity is established.
- Authorisation. Upon successful authentication step, user is allowed to perform certain action conform to the security policy.

There are two participants in the authentication process: the authenticating authority and the authenticated party. According to the Jericho commandment number 6 "All people, processes, technology must have declared and transparent levels of trust for any transaction to take place", therefore not only a user will have to authenticate himself, but also a network device that is being used to access the resources. Services and authenticating authorities must be authenticated as well. There are several requirements that an authentication system must meet. Review of the authentication system products falls out of the scope of this research. It is important to set criteria for the evaluation of existing system. By doing this one can determine if his architecture can be easily extended to a de-perimeterised network. In some cases revision or upgrade may be required.
Phillip Windley in his book "Digital Identity" sets the basic requirements34. These requirements are extended for the usage in de-centralised network.

- **Support for federation or user centric design.** It is the most important requirement for Jericho project. Support of the last two models is a must for an authentication system. It must be also scalable enough to support all the external connections of an enterprise.
- **Practicality.** An authentication system should be non-intrusive and easy to use. From the user point of view it is inconvenient to authenticate for every resource on the network. From the enterprise point of view an authentication system should be cost-effective and easy to deploy.
- **Appropriate level of security.** An authentication system should be able re-assert itself if authentication level is insufficient. Multiple factor authentication should be supported. User should also be able to choose which factors he prefers.
- **Locational transparency.** User must be able to authenticate himself regardless his physical location. Location parameters will be used for authorisation purposes.
- **Procotocol insensitivity.** Systems should be able to interoperate regardless the transport protocols they use.

[33] Renaud, Karen.,'Evaluating authentication mechanisms'. In Security and Usability: designing secure systems that people can use. O'Reilly 2005. chapter 6
[34] Windley, Phillip J. Digital Identity. O'Reilly, 2005

- **Appropriate level of privacy.** Authentication system must comply with the privacy regulations of the company.
- **Reliability.** Authentication system is a crucial element of enterprise architecture. Appropriate redundancy is required in order to achieve a high availability level.
- **Auditability.** All transactions in the authentication system should be audited and retained for a required period of time.
- **Manageability.** User accounts should be easily manageable.
- **Support for multifactor authentication.** To receive sufficient assurance in user identity and meet the requirements of the authorisation policy, an authenticating authority may require additional factors for the verification.
- **Device authentication.** Some authorisation policies may require additional device authentication in order to verify that the user is using an appropriate device for information access.
- **Additional parameters.** Authenticating authority must be able to include additional parameters which can be transported together with the claims and assertions. Such parameters may include: GPS data, security status, etc.

An enterprise which changes its infrastructure according to a de-perimeterisation concept is advised to analyse whether its authentication system is suitable and all requirements are met.

User authentication

Credentials that are used in the authentication process are unique and belong only to a certain user. There is variety of credentials that can be presented to the system. The more credentials are present in the process the more secure it is. This is also known as multiple factor authentication.

The three factors of the authentication are:

- Something you know (Memometrics and cognometrics)[35]
- Something you have
- Something you are (Biometrics)

Something you know

Memometrics is something a person can memorize. It can be a password, passphrase or a pin code. Cognometrics is something a user can recognize. It relies on the fact that the human ability to memorise a picture is very powerful.

A user is authenticated by recognising a certain picture (visual memory) or placing/drawing a certain image on a grid (visio-spacial memory). Cognometric systems are considered to have very strong security. Playback attacks are excluded because images may change their location every time authentication is performed. Parties are mutually authenticated as well. By recognising images a user can be assured in validity of the other side.

[35] Nomenclature introduced by www.realuser.com

Something you have
Usually it is a hardware or software token that a user has in his possession. The token is used to generate a one time password that can be accepted by the authenticating authority. A token is considered to be a very secure way to authenticate a user, but its biggest disadvantage is that it always has to be carried by its owner. A token has an additional protection and usually a PIN is required in order to use it.

Something you are
Biometrics is the oldest form of authentication. You recognise the voice of your friends and their faces. A bank checks your signature before they process the money transaction. There are multiple possibilities to identify a person: palm, ear, hand geometry, iris, retina, fingerprint, face, voice, signature and many others. One kind can be more or less secure than another. One should choose carefully before implementing it. Costs and security should be balanced. Biometric parameters are highly reliable and are very difficult to forge, but when biometrics are compromised, they remain compromised for the rest of life. It is impossible to issue new irises or fingerprints. "Biometrics are powerful and useful, but they are not the keys" says Bruce Schneier.[36] This is why biometrics are not advisable for the global usage. Locally (within single enterprise) biometrics can be used as an addition to the current authentication mechanisms as multifactor authentication.

There are multiple factors that can be used in the authentication process. The more factors are present the more assurance receives the authentication authority in the user identity. Biometrics is the strongest way to authenticate a user, but once it is compromised it is impossible to recover it. This is the main reason Bruce Schneier does not advocate the global usage of biometric authentication.

Network device authentication

After the user identity has been established, some policy may require authentication of the device that the user uses to access the network. There are several possibilities for this[37]:

- **MAC address**. MAC address is the hardware address which is present in all network cards. This address is unique and it consists of two parts: manufacture and unique number. MAC addressing is used in the communication process at the second layer of the OSI model. This hardware address can be used to identify a network device on a local network. Several security mechanisms are based on it. For example a MAC filter on the wireless access point. Unfortunately all operating systems allow changing MAC address using software means.
- **Trusted Platform Module (TPM).** Most of the modern computers are shipped with a hardware chip that allows device authentication. The chip contains unique information that would allow secure communications. It can also be used for

[36] http://www.schneier.com/essay-019.pdf
[37] http://www.xml-dev.com/blog/index.php?action=viewtopic&id=243

cryptographic and DRM goals. This chip has failed to receive global acceptance and on the most systems it is disabled for privacy reasons. Desktop, notebook and tablet PCs with TPMs are available from Dell, Fujitsu, Gateway, HP, Intel, Lenovo, Toshiba and others. Trusted servers also have started shipping.[38]

- **Software based device authentication.** Windows Security Identifier, installed software or a license key can create a unique signature that could identify a system on the network. Of course most of the factors are variable, but windows license key doesn't usually change very often. Additional software is required to create the signature and process the authentication requests.
- **Certificate based.** PKI based infrastructure can be easily developed and deployed. Certificate can be installed on every device through user self-enrolment process. In Microsoft Active Directory Domain enrolment it is an automated process that doesn't require user intervention. PKI infrastructure would require trust relationship between parties. See "Authentication and trust" chapter for more information.
- **IP address based authentication.** Every device on the network has a unique address which allows layer 3 communication with other devices. In version 4 of the TCP/IP protocol the number of the public IP addresses is almost depleted and it would be impossible to supply every network device in the world with a layer 3 address. Another barrier for using the IP address as an authentication method is that it can be easily spoofed. IPv6 addresses solves these two problems, but a soon implementation of this protocol is doubtful.

The only easily deployable and reliable solution for a device authentication is to create a global PKI infrastructure. Another solution that would solve the device authentication problem is TPM but means for user consent and control should be created first. User should be able to disable or control its usage. The disadvantage of this solution that in case of the malfunctioning a new module cannot be deployed that easily as for example an instalment of the new certificate.

Authentication and trust

Authentication heavily depends on trust. IBM gives the following definition of trust as following "Trust is the expression between parties that one party to a relationship will believe statements (claims) made by another party; it is based on evidence – history, experience, documents, etc. – and personal risk tolerance."[39] In a centralised model a symmetric trust is established between an authenticating authority and a relying party. Identities issued by this authority are accepted by the relying party. This can be compared to an enterprise building. When a guest enters the building he acquires a visitor's card. By presenting the card he may be admitted to the certain areas. Security guards trust his credentials because it is issued by their colleagues at the reception. In this case we can

[38] https://www.trustedcomputinggroup.org/faq/TPMFAQ/

[39] http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-fed/WS-FederationSpec05282007.pdf

trace the rlationship between the issuing authority and the relying party. The federation and the user-centric models adopt a different kind of trust – asymmetrical trust. A real world example would be showing a passport at the border control. The officer who checks your passport does not have any relationship with the issuing authority, but he still trusts the issuer. In the virtual world a relying party should be able to trust the claim or assertions of an authority it has never heard of before. Mutual local trust that exists in the centralised model will disappear in federation and user centric models. In federation and user-centric model there are several trust structures possible. How trust is being managed and established is described in detail in 'Jericho in Depth...Trustbroking Services by Adriaan Bruning – Capgemini 2008'. The possible trust and collaboration models are described below. It is important to give insight of the identity flow and how it can traverse multiple trust links. Trust models are very similar to PKI infrastructure.

Hierarchical model

Hierarchical model implies analogy to the current DNS model, where "." is the root issuing facility. Underneath we see different organisations, which in their turn issue identities to the underlying companies. The division of the second layer maybe grouped according to a geographical location or function.

An example of this model can be a government, which issues en identity. This single identity will be presented to the multiple parties.
This model has its pros and cons:

+ Easily manageable infrastructure
+ Identity must be approved before it is issued
+ Global security policies

- A non-profit organisation must exist in order to manage the hierarchy
- No multiple identities are possible
- A single organization is in charge of the identity management
- Partial implementation is difficult
- Single point of failure
- Policy imposed by the top organisation must be implemented by the underlying organisations.

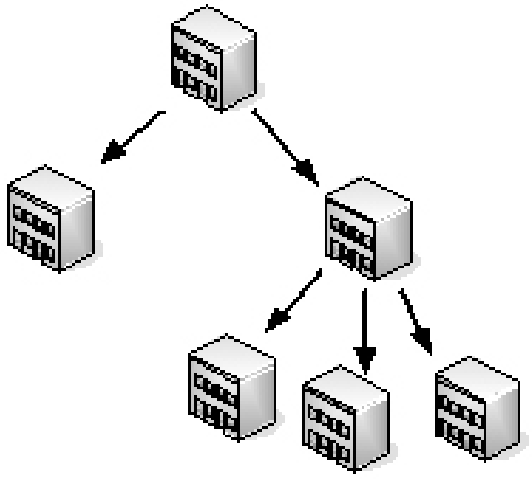


Figure 18 - Hierarchic trust infrastructure

Flat model

Flat infrastructure implies that every single identity provider has its own policy to work with. Identity providers may also participate in the identity hierarchy and be approved by one single company as for example VeriSign. Every company may ask for such approval. For example it is essential for a bank to be verified by a single trusted organisation.

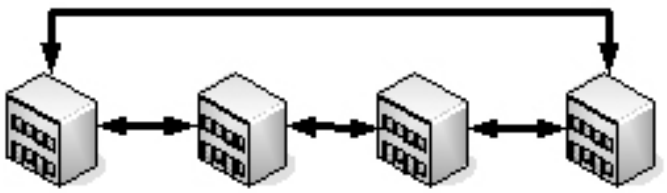
Figure 19 - Flat trust infrastructure

Companies lay trusts to other organisations but they all stay on an equal level. A trust may be transitive or intransitive by nature. Each company defines security policy and its trust nature according to their needs.

Pros and cons of this model:

+ No single company is in charge
+ Policy is locally defined
+ Flexible and scalable infrastructure

- Manageability issues
- Difficult to identify trust relationships



Hybrid model

Organisations participating in a trade or belong to the same business branch may cooperate with each other and create an authority on top. The top authority is the most trusted authority and issues identity to organisations. The organisations in their turn issue identities to the subsidiary companies and employees. An employee from one company is trusted by another company. In some cases an identity may also be issued by the top authority. This model has advantages and disadvantages of the both pervious models.
Companies choose the most suitable model and implement it upon the agreement with other organisations. It depends whether an enterprise wishes to collaborate and what is the nature of this collaboration.

Trust is the foundation of the collaboration. User centric and federation models deploy a different kind of trust – asymmetric. By understanding the trust infrastructure one can trace the identity flow between the domains. There are three possible collaboration models: flat, hierarchic and hybrid. Flat is the most simple one and it enables elementary communication. Hierarchic model is the complicated collaboration model, where the most trusted authority is on the top of the infrastructure. Hybrid model is a mixture of the both of them.

Additional authentication parameters

Security and authorisation policies may require additional parameters to be transmitted together with the identity data. They can be transported in the assertions or exist as separate claims. This additional information may describe user location, network device security status, operating system and environment.

User location

This additional parameter may be required for location based authentication and authorisation. Some information may be read only within the building and may not be accessed by users which are authenticated and accessing the network through a VPN connection. Another example: if a user is registered when he enters the building and according to the system is still inside. Why should he be able to access data through VPN?
User location problems can be solved by fusing physical access and directory authentication systems. Another possible solution for this problem is GPS information transmission to the authorisation service. On basis of that is the decision made whether user is allowed to access data.

Network device security status

A confidentiality policy may require that the user device security status should be verified before accessing the data. Customisable policy may check antivirus signatures, system updates and firewall status. Information may be transferred to this device but precaution should be taken to make sure that it won't leave it in unauthorised way (information leakage). Security status update frequency may be regulated by the security policy of the service or data self. Thesis "Jericho Forum Project: Endpoint Security and Authorisation" by

Leon Teheux gives more insight into the technical solutions which can perform network preadmission scans and risk mitigation.

Operating system and environment

User's operating system and installed applications may be a very important factor for the authority that makes an authorisation decision. If the policy differentiates fully capable operating system (e.g. Microsoft Windows XP, Ubuntu linux) from an embedded one (e.g. Symbian), then this information should be transmitted as well. Another factor is environment and installed applications. Some DRM and information leakage prevention solutions require certain software or agents to be installed on a networking device. Whether they are installed and enabled is the task of the security scan. This information can be retained in the user attributes and transmitted to the relying party upon request.

Some domains may require additional parameters to be sent with user assertions. User location is needed for the location based authorisation. Network device security status and operating system / environment parameters are needed for the risk analysis.

Authentication process

SAML 2.0 authentication process

SAML authentication uses assertions as proof of the user identity. Here is a detailed description of the process. It is important to stress that this is a very general case and there are multiple variations possible.

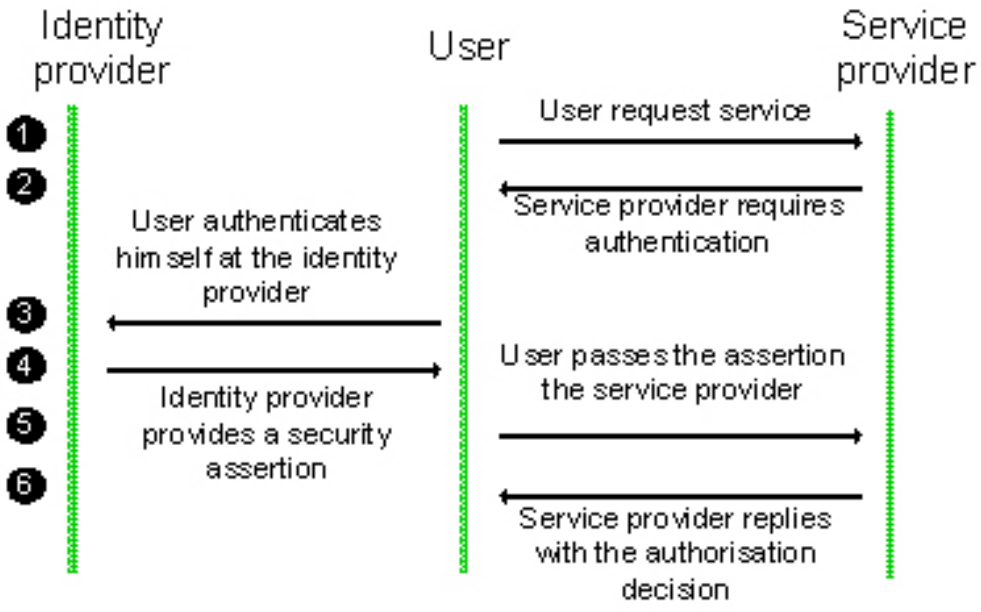


Figure 20 - SAML authentication process and identity flow.

1. User makes a request to a resource.
2. Source request user's identity.
3. User goes to the identity provider and request identity federation.
4. Authentication authority provides user with a security assertion.
5. User passes the provided assertion to the service provider.
6. Service provider makes and authorisation decision and replies to the user.

As described earlier there are protocols for an assertion request and assertion response. There are also different types of assertions. Two types are important within Jericho forum research framework: authorisation statement and attribute statement.

Authentication statement asserts that a principal is authenticated at the specified time with the specified means. SAML differentiates levels of confidence they can put in an assertion which allows to make a distinction of the certain users. SAML does not prescribe a single technology or protocol by which authentication authority may issue identities to its users. There is variety of different mechanisms which can be engaged in order to issue a high assurance security assertion which may be required for a higher level authorisation. The authentication context is an addition to the authentication statement and define means by which a principal is authenticated. This information may be used by risk management processes to make authorisation decision.

SAML allows the following information to be delivered to the relying party[40]:

- The initial user identification mechanisms (for example, face-to-face, online, shared secret).
- The mechanisms for minimizing the compromise of the credentials (for example, credential renewal frequency, client-side key generation).
- The mechanisms for storing and protecting credentials (for example, smartcard, password rules).
- The authentication mechanism or method (for example, password, certificate-based SSL).

A relying party may request additional information and a principal will have to re-authenticate himself with different methods in order to receive information that requires this kind of measures.

Attribute statement may potentially carry information relating to the host security. After the pre-admission scan and user authentication this information can be dynamically inserted into the user attributes. When user requests federation his security information will be

[40] http://docs.oasis-open.org/security/saml/v2.0/saml-authn-context-2.0-os.pdf

delivered in the attribute statement assertions. Also additional user information such as location and installed applications may be transported this way.

Conclusion
SAML 2.0 may deliver additional information to the relying party. Based on this information a more careful authorisation decision can be made. Even if multiple factor authentication was present, a user may not be authorised to access certain sensitive information.
The attribute statement can be utilised for the transfer of the additional information such as host security status, user location and environment/applications. This information can be dynamically updated every time a scan is performed.

Microsoft Identity Metasystem authentication process
The user centric authentication process is slightly different from federation authentication. The major difference is the user agent. It operates as a medium between relying party, user and the identity provider. It also collects the needed claims (assertions) and presents them to the relying parties. An authorisation decision is based upon the authorisation context claims requirements and the presented claims. Claims can also contain rich additional information in case it is required by the relying party. Interesting point here is that claims can also be based on other claims. Data authorisation policy contains claims that are required to access it. User agent manages claims and identity information and it also shields user from the phishing attacks. By combining claims and sets of claims one can establish a refined security policy. For example one can choose which authenticated authority token he uses in order to access the resource or a resource may require multiple tokens in order to grant access.

On figure 21 you see the protocol drill down to see how the client communicates with the resource. The presented scheme shows the simplified communication of identity metasystem set of protocols.

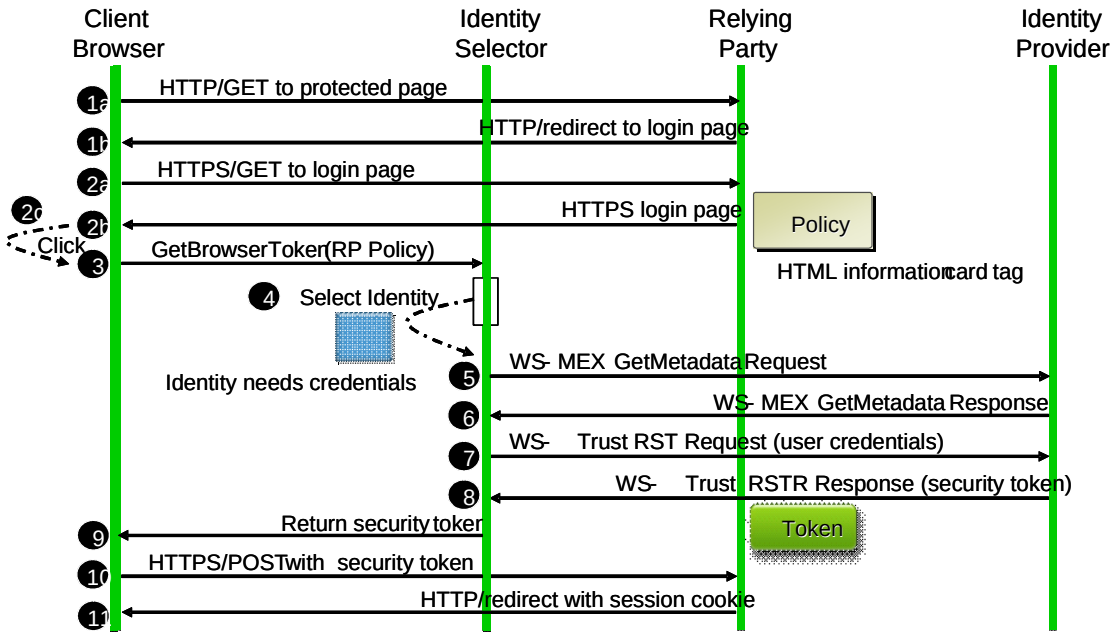


Figure 21 - Identity Metasystem authentication process

Process:

1. Steps 1a - 2b. The user communicates with a relying party and requests data or service.
2. Step 3. The browser passes the request together with the policy of the relying party to the identity selector (user agent).
3. Steps 5 -9. The user agent analyses the requirements and chooses the identity providers that can satisfy these requirements. After that it acquires the security tokens from these authorities and returns them to the browser.
4. Steps 10 – 11. The browser passes the security token to the relying party and receives the requested services or data in return.

Identity Metasystem authentication offers protection from the phishing attacks. User agent shields user and provides authentication of the relying parties. Mutual authentication is one of the requirements of Jericho Forum. Rich tokens can be used to specify the authentication type, but the variety is not as rich as it is offered by SAML.

Conclusion

There are number of requirements for an authentication system which need to be met in order to build a de-perimeterised network. Some of them like support for federation and user centric models are essential. It also has to be able to perform multifactor authentication which allows an authorisation authority to discriminate some users. Some information on the enterprise network can be classified as sensitive and a user may be required to be re-authenticated with the additional factors.

Biometrics is the strongest type of the authentication but it is not advised for global usage. Once it is compromised it cannot be restored. This is the main reason that the usage of biometrics should minimised by the local authentication only.

Device authentication can be required by the authorisation policy as well. There are possible solutions for that, but until now none of them has proven to be the ultimate one. Authentication based on MAC and IP address can be easily spoofed. TPM and certificate based systems can potentially be used in the global infrastructure. TPM offers more secure but also more expensive and inflexible solution. Certificate based solution can easily be deployed in the current infrastructure.

Trust lays the basis for collaboration. It is important to understand it in order to determine how identity can be federated between the organisations. Through the trust infrastructure we can trace the identity back to its originating domain and request the authentication there.

Additional information about the user location, host security status and user environment can be requested by the relying party. This data can be analysed for the risk calculation. User may have sufficient authentication level but he still may not be authorised to access the information. This may prevent users from accessing sensitive data from potentially insecure devices or it can be used for the location based authorisation.

SAML and Microsoft Identity Metasystem offer similar possibilities for user authentication. The biggest advantage of SAML is a broad authentication context, it also offers possibilities to transfer the user attributes. Additional authentication parameters can be dynamically inserted and transported as the user attributes. Microsoft Identity Metasystem offers rich claims to identify user authentication type. Another advantageous feature of this system is the user agent, which allows mutual authentication between the relying party and the user. Mutual authentication is one of the requirements of Jericho Forum.

6. Accounting

"Repetition builds continuity, Continuity builds history, History builds Identity."
- Roshan Samtani

European Union is a good example of federation. Every country is a security domain with its own policy. They all participate in identity exchange: citizens of every state may freely move and settle within other states of the union. But the citizens' history and reputation of one state is not known to the authorities of another state. They also do not trace the citizens' movement (Privacy issues). When a person leaves a country he disappears for the local police.

The same situation exists in the virtual world. We free the identities by participating in federation, but means to exchange information about that identity are not defined. Accounting is a crucial process for a modern enterprise. Organisations have established security regulations about how data and what kind of data is collected. Information which is generated and kept in logs is used for multiple purposes. How it can be used it discussed later. What important is to realise that this information always stays within the boundaries of the enterprise. But this data which is collected about a subject may present a very high value, not only for the local domain but also for the collaboration partners. Jericho forum commandment nr 8 "Authentication, authorisation and accountability must interoperate / exchange outside of your locus / area of control" requires that auditing data can be exchanged between the security domains. When you collect data about an identity which was federated into your security domain, you can acquire information only after it was authenticated. There is no possible way you may know which resources were accessed by this person in other domains. Unless, of course, other security domains share this data with you so that you can receive entire information and make a more careful decision whether you want to trust or not trust this identity in the future.

My intention is to show in this last chapter that information exchange is crucial for risk assessment and trust management, but not the creation of the military surveyed Internet. By no means is it suggested that this kind of personal information should be shared to everyone who requests it. Data is provided only to the trusted parties and users should be informed about the monitoring process.

Accounting and auditing

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. Let us discuss the major three.

- **Network performance analysis.** Network and system administrators analyse logs in order to evaluate system / network performance and utilisation and to proactively manage it. For example capacity of the network should be extended in time so that user experiences no difficulties accessing resources.

- **Financial purposes**. Financial department uses logs to generate a usage report and charge users for the use of the certain services. Also costs management within project may use accounting to allocate the expenses.

- **Auditing.** Auditing refers to the monitoring of user activity. Auditing can help to determine gaps in the security and policy violations. In case unauthorised access has already been established to the network resources or the network has suffered a cyber attack. Security professionals may use audit logs to collect evidence for the forensic investigation. Chains of events may be recreated to identify the person who intentionally or unintentionally has committed a criminal offence and hold this person liable for his actions. Liability may be criminal, civil or administrative. Auditing can also help to enforce the security policy of the enterprise. By monitoring the user actions one can easily determine the illegal activity and warn the user.

Accounting policy must conform to a company's security policy which is defined by the company management and the current security standards. Retention period of the log files may depend on the certain factors, such as established principles and security policy. Security policy in most cases must comply with the regulations of the country, within which jurisdictional boundaries the company is situated. From the three distinct purposes of accounting, auditing is the most valuable for Jericho forum project. The main purpose of is it to monitor user activity and define breaches in the security policy. Part of the auditing data can also be used to perform reputation analysis and to define or in the future redefine trust relationship with the user or an enterprise.

Exchange of the information between autonomous domains participating in federation is very important. When identity crosses the security boundary no information is available about it. Accounting is the process of keeping track of the online user activity. It can be used for multiple purposes. The security purpose is known as auditing. It refers to the monitoring of the user activity. When this information is exchanged between the domains it can help to build a stronger collaboration relationship.

Auditing requirements

As defined in the previous chapter, identity audit data has to be exchanged. But what is this data and what is needed to audit in order to create a clear picture of the user activities? The identity lifecycle establishes the logical framework for the auditing requirements. For each step within the cycle we can define auditing rules which comply with the security policy. It is very important to note that some countries have very strict privacy laws and acquiring certain data which can be traced to a person, is strictly prohibited. On a centralised network model it is very easy to perform audit and trace user activities. Let us have a look at the general activities per cycles which need to be audited.

Provision

- Act of identity issuance
- Request for identity issuance
- Requestor of the identity issuance
- Performer of the identity issuance
- Approver of the identity issuance
- Resources that the identity is eligible to access

Propagation

- All the systems where identities are propagated to

Use

- All authentication requests
- Resources accessed by the identity
- All access granted/denied responses

Maintenance

- Identity mutations
- Performer of the mutations
- Requestor of the mutations
- Approver of the mutations

Deprovision

- Request for identity termination
- Requestor of the identity termination
- Performer of the identity termination
- Approver of the identity termination

Enterprise may advance the auditing of the activities according to their needs and perform it as extensively as it is required. In opposite of that some may want to reduce unnecessary auditing and protect the privacy of the users.

The requirements for auditing are logically defined by the identity lifecycle. All activities may be logged. Company may also extend this list or in opposite reduce the audited events.

Distributed auditing architecture

In the federation and user centric models the identity provider and the resource may be situated in independent security domains and because of that one cannot collect sufficient audit information as easy as in the centralised model. Data exchange between the security domains is needed. Local audit processes supply information to the exchange process and receive external information in return.

Figure 20- Interdomain audit data exchange

Let us have a closer look at the accounting processes. They are divided into local auditing and interdomain audit data exchange.

Local audit[41]

Local auditing assumes that two distinct security domains cannot share their auditing data and each of them keeps track of the activity on its own side. Identity providers monitor provisioning, maintenance and deprovisioning operations. Relying parties monitor the use of identity in their domain. If this data doesn't leave the security boundaries it delivers maximum privacy to the users. Audit information collected in this way can be more than adequate for an enterprise. Identity provider monitors issuance of the identities and changes made by the administrators or users self. Information about the usage of the identity may not be relevant to them. Relying parties may only be interested in information of which resources are accessed by the external identities and whether there are violations of the policies.

1. **Applications and services audit trails.** Applications and services are dispersed over the whole security domain. Each of them has its own audit trail. This data is collected locally. Every application and service maintains its own audit policy. For ease of management they all can be set to default of maximum information auditing.

[41] OpenXDAS specifications were taken as basis of the distributed internal audit process. See http://openxdas.sourceforge.net/

2. **Audit input**. Auditing API's collect information flow from the entire security domain through import or submission of the audit trails.
3. **Common format.** Common format process ensures that different kinds of logs can be transformed into the standard format. This ensures interoperability not only within the local domain, but also on the interdomain level.
4. **Event discrimination**. It is a process which determines the event disposition and filtering. The pre-selected events are forwared to the following process, the rest is nullified. This process is regulated by the security, auditing and privacy policies of the local domain.
5. **Central repository**. Events that have passed the discrimination service are placed in the central repository. This is a database where logs from the local and external domains are retained. Retention period is also regulated by the security policy.
6. **Audit analysis.** At this point all audit data is submitted to the analysis application. It reveals violations and calculates risk.

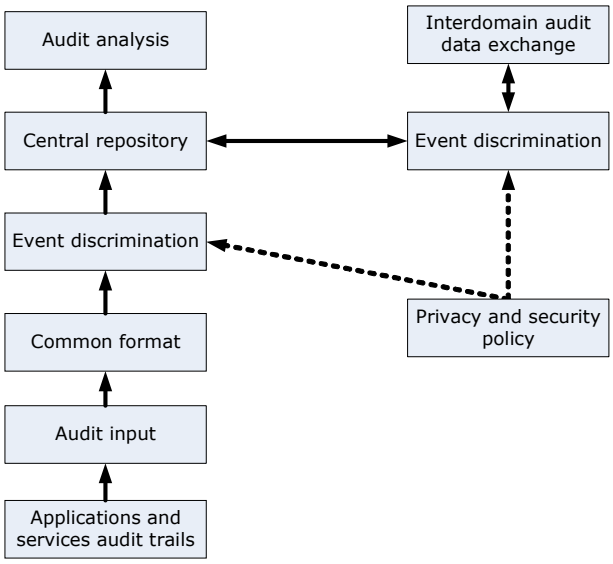


Figure 23 - Audit process (Detailed view)

Interdomain audit data exchange

When an enterprise requests more audit information about the identity, the means should exist to deliver this data to the requestor. Interdomain auditing process delivers the information which can be required by security policy from another domain. Enterprises agree upon information exchange and underlying protocols. Policy agreement defines what information can be exchanged. For example an identity provider may request the relying party to deliver the detailed information of the resources usage by an identity. Another case may be an independent third party which provides a reputation service for both identity

providers and relying parties. A reputation service can for example assist enterprises that do not have trust relationship established yet, but they want to form an adequate image of the company they want to collaborate with. Reputation data is calculated through algorithms and should not contain any personal data.

1. **Event discrimination.** Before audit data is submitted for transport it is evaluated against the privacy and security policy of the local domain. Events which are not authorised to leave the domain are nullified. Events that enter the domain are checked against the auditing policy whether they are needed. The required events pass to the central repository, where they are added to the local events.

2. **Interdomain audit data exchange.** This process exchanges data between local and external domains. Data is encapsulated into the secure transport protocols and transmitted to its destination. Data requests also processed here. The query is passed to the central repository and it returns data in response. This process accepts only queries from the trusted domains with which they have established an agreement of the data exchange.

The audit information which is exchanged between parties may be personal or impersonal in nature. Organisations situated in the countries with the strict privacy laws or whether they just want to protect their users can reveal general information with no personal information in it. Enterprise chooses the most suitable solution: implementation of the local audit or data audit exchange. The decision is based on the following factors: expenses, level of collaboration, security and privacy policy. If an enterprise chooses the distributed consolidated local auditing on the first place, they may want to participate in audit data exchange and deliver or receive information which is collected in the local or external security domains later.

Distributed audit architecture consists of local domain audit and interdomain audit data exchange. Local audit data consists of several processes. Each of them is responsible for one of few functions. Data is retained in the central repository, from where it can be requested and transferred to other domains. It is important to ensure that data is transformed into a common standard which can be understood by both domains. Event discrimination process filters out the unneeded events and protects the privacy of the users.

7. Conclusions and recommendations for further research

More and more organisations start to realise the real value of de-perimeterisation. Enterprises and individuals require a simplified and secure architecture for the communication and collaboration. The current concept of identity management is inefficient and cumbersome. There is need for identities which can be taken outside the perimeter of the local security domain. Federation and user centric identity models provide this possibility. Federation model can be built on the current infrastructure and it provides means that can federate the local identities and make them usable in other domains. Unilateral or bilateral federation agreements are required in order to enable it. Unilateral agreement is mostly used by a relying party (service provider) that trusts any issuer of the identity and everybody is allowed to use its services. Bilateral means that both parties have to sign a collaboration agreement in order to mutually accept the identities. There are multiple identity systems that may be used for identity federation. At the moment the market is lead by SAML 2.0. It is easily extensible and it delivers important security features which are essential for the Jericho Forum concept. The user centric model truly erases the domain boundaries as it is envisioned by the Jericho Forum. Identity management is for its biggest part shifted to the user side and makes him responsible for his identity. User centric model is the true reflection of the real world situation where a person actually manages his identities. The concepts of self issued identities and card revocation are also supported there. Unfortunately the user centric model is still partially conceptual and needs further standardisation. In my personal opinion user centric model is the following step in the evolution of the digital identity, but it needs time to be crystallised and accepted. There are few identity systems solutions available which can be used to implement the user centric model. Open ID is a more suitable solution for global usage such as Internet. Microsoft Identity Metasystem is still in beta testing and needs time to prove itself before it can be deployed in an enterprise.

Before introducing a de-perimeterised network concept an organisation has to verify whether the authentication system meets the requirements. The most important requirement is that it should support federated and user-centric models. Another requirement is that user should be able to be authenticated with multiple factors in order to issue a high assurance claim. Some authorisation authorities may require the additional information about the user (location, security status, installed applications). Some data may be classified as very sensitive and require the end point security to be up-to-date and certain applications to be installed. This can prevent information leakage from a network device. Security status information can be dynamically inserted as a user attribute and transported to the requesting authority. Resources in the de-perimeterised network can be dispersed over multiple security domains. It is essential that auditing data can be exchanged when it is required. Data exchange should conform to the privacy and security policy of the local domain and should only be available to the trusted parties.

Auditing architecture consists of two separate processes: local security audit and interdomain audit data exchange. It is important to bring the auditing events to the common

standard. This facilitates communication between independent domains. The data can also be used by the third party service which offers reputation information. It may help to build a collaboration trust between enterprises and individuals which have none or little information about each other but want to do business together. De-perimeterisation enables cost effective and very secure environment for businesses and individuals. The current architecture does not meet the requirements the modern businesses demand from the IT solutions. Over more than 1000 years we have established principles, traditions and practices which govern our physical world. The digital world is completely different from that. Jericho Forum has made the first leading step towards translating these underlying basics so that they can accommodate our needs in the Internet just as easily and securely as they do in the real world.

Recommendations for the further research

There are few topics which are partially covered by this research but require further scrutiny.

Relying party authentication

Jericho forum requires all parties of the communication process to be authenticated. Microsoft Identity Metasystem does this per default. User agent authenticates the relying party and shields the user from phishing attacks. In the federated identity systems it is still has to be implemented. If the relying party is not authenticated it can potentially bring user identity data at risk.

Audit data request

Local audit processes can be implemented with OpenXDAS. It is an open source distributed audit software. It can also be used to transport data between the domains. The implementation of the audit data request and audit data reply remains open.

Additional security information

Data collected through the security scan or acquired from other devices (e.g. GPS) must be transported to the authorisation authority. Data can be transported directly or be dynamically inserted as a user attribute. Technology is needed that can import that data from the security appliance and user agents and update the corresponding field of the user profile.

References

Books

1. Renaud, Karen. 'Evaluating authentication mechanisms'. In Security and Usability: designing secure systems that people can use. O'Reilly 2005. chapter 6
2. Windley, Phillip. Digital Identity. O'Reilly 2005.
3. Adriaan Bruning. Jericho in depth...Trust broker services, Capgemini 2008
4. Marle, Remco van. Jericho Forum Project: Data Classification. Thesis document, Capgemini 2007
5. Stan, Alina. Jericho in depth...Secure Communications Capgemini 2008
6. Teheux, Leon. Jericho in depth...Authorisation and Endpoint Security. Capgemini 2008

Electronic sources

7. http://biz.yahoo.com/prnews/070507/sfm045.html?.v=92
8. http://blogs.sun.com/hubertsblog/date/20070302
9. http://cardspace.netfx3.com/content/introduction.aspx
10. http://eid.belgium.be
11. http://encarta.msn.com/encnet/features/dictionary/DictionaryResults.aspx?refid=1861619974
12. http://identity20.com/media/OSCON2005/
13. http://kronkltd.net/blog/?p=764
14. http://openxdas.sourceforge.net/
15. http://www.capgemini.com/about/
16. http://www.digitalidworld.com/modules.php?op=modload&name=News&file=article&sid=26
17. http://www.ibm.com/developerworks/webservices/library/specification/ws-secpol/
18. http://www.identityblog.com/stories/2005/05/13/TheLawsOfIdentity.pdf
19. http://www.linux-mag.com/id/3353/
20. http://www.oasis-open.org
21. http://www.xml-dev.com/blog/index.php?action=viewtopic&id=243
22. https://www.trustedcomputinggroup.org/faq/TPMFAQ/

23. www.opengroup.org
24. www.opengroup.org/jericho
25. www.realuser.com
26. www.searchvoip.com
27. www.sxip.com

White papers

28. http://docs.oasis-open.org/security/saml/v2.0/saml-authn-context-2.0-os.pdf
29. http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-fed/WS-FederationSpec05282007.pdf
30. http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-fed/WS-FederationSpec05282007.pdf
31. http://www.oasis-open.org/committees/download.php/13525/sstc-saml-exec-overview-2.0-cd-01-2col.pdf
32. http://www.oasis-open.org/committees/download.php/20645/sstc-saml-tech-overview-2%200-draft-10.pdf
33. http://www.opengroup.org/jericho/commandments_v1.2.pdf
34. http://www.pingidentity.com/download/show/289
35. http://www.projectliberty.org/index.php/liberty/resource_center/tutorials/liberty_quick_start
36. http://www.projectliberty.org/liberty/content/download/1028/7146/file/liberty-idsis-pp-v1.1.pdf
37. http://www.projectliberty.org/liberty/content/download/1297/8256/file/liberty-idwsf-disco-svc-v1.2.pdf
38. http://www.schneier.com/essay-019.pdf

List of abbreviations

AAA – Authentication, Authorisation and Accounting

BeID – Belgium Electronic Identification

DNA – Deoxyribonucleic Acid

DNS- Domain Name System

HRM – Human Resource Management

HTTP – Hypertext Transfer Protocol

ID – Identification

ID-FF – Identity Federation Framework

ID-SIS - Identity Services Interface Specifications

ID-WSF - Identity Web Services Framework

IP – Internet Protocol

IP- Identity Provider

IT – Information Technology

MAC - Media Access Control

PIN – Personal Identification Number

PKI- Public Key Infrastructure

RP – Relying Party

SAML – Security Assertions Mark-up Language

SOAP – Simple Object Access Protocol

SSL - Secure Sockets Layer

STS – Security Token Service

TCP - Transmission Control Protocol

TPM – Trusted Platform Module

TTU – Telecom, Travel and Utilities

VPN – Virtual Private Network

WS – Web Services

XDAS – Distributed Auditing Service

XML – eXtensible Mark-up Language

List of definitions

The definitions are borrowed from the following resources:

1. http://identityaccessman.blogspot.com/
2. http://msdn2.microsoft.com/en-us/library/ms951235.aspx

Access – The ability to use a resource or a service.

Account – An instance of an Identity. An Identity may have multiple Accounts. Usually associated with a single computer application or platform, but also applies to such things as bank accounts, utilities and telephone accounts.

Attestation – The confirmation, corroboration or formal acceptance that something is correct.

Attribute – A type/value pair of information related to an Entity or Identity. It may be shared (e.g. nationality), or unique (e.g. DNA). A combination of attributes may be sufficient to satisfy an assertion. Usually a value in an identity repository (directory or database) collected directly or indirectly through registration, enrolment or access control.

Authentication – The process of establishing an Identity to be used in a particular instance, by verifying an assertion (e.g. claiming to be the owner of a set of credentials).

Authorisation – What the Identity can do, in a given instance, as a result of proving an assertion.

Biometric – A physical trait or behavioural characteristic that can be used for the purposes of identification or verification. A good biometric should be unique to an individual, stable over time, quick and easy to present and verify, and not be easily duplicated by artificial means.

Claim - A claim is a declaration made by an entity (e.g. name, identity, key, group, privilege, capability, attribute, etc).

Credential – The private part of a paired Identity assertion (user-id is usually the public part). The thing(s) that an Entity relies upon in an Assertion at any particular time, usually to authenticate a claimed Identity. Credentials can change over time and may be revoked. Examples include; a signature, a password, a drivers licence number (not the card itself), an ATM card number (not the card itself), data stored on a smart-card (not the card itself), a digital certificate, a biometric template.

Digital Signature – An electronic signature that can be used to authenticate the identity of the sender of an electronic message or the signer of a digital document, and to ensure that the original content of the message or document that has been sent is unchanged.

Directory – a hierarchical repository used for authentication and/or identity management. Usually based on the X.500 standard and LDAP protocol.

Encryption – The conversion of clear text (readable data) into a form called cipher text that cannot be easily understood by unauthorised people or systems, by using cryptographic keys.

Enrolment – The process of adding a Permission to an Identity. It may result in the issuing of a new identity or an additional account. The link between Registration and Enrolment must remain unbroken.

Entity – anyone (a natural or legal 'person') or anything with a separate existence that can be characterised through the dimension of its attributes.

Event Logging – The recording of details of an end-to-end enterprise-wide process, for audit purposes. It should have the ability to give a single picture of the actions of any identity over time.

Factor – The fundamental classification of credential types. There are actually only three factors: what you 'know', what you 'have', and what you 'are'. Combining two, or three, into a multiple-factor solution is a means of stronger authentication. There are suggestions from time to time of new factor classifications such as 'what you do' or 'where you are', but they always resolve into the basic three.

Federated Identity – A shared Identity and/or authentication, as the result of federation by either the Entity or by two or more organisations.

Federation - A federation is a collection of realms/domains that have established trust. The level of trust may vary, but typically includes authentication and may include authorisation.

Federation – A method of linking together the Identities of an Entity, to provide shared services as a matter of convenience, efficiency and trust.

Group – A set of one or more Identities that can be authorised under one Rule. An Identity may belong to zero, one or more groups.

Identity Management – Formal standardised enterprise-wide or community-wide processes for managing multitudes of Identities.

Identity Provider - Identity Provider is an entity that acts as a peer entity authentication service to end users and data origin authentication service to service providers (this is typically an extension of a security token service).

Kerberos – (Greek mythology: the three-headed dog that guarded the gates of Hades). An authentication service that issues a ticket-granting ticket and a one-way hashed session-key (for encryption), stored in a cache. It requires the continuous availability of the Kerberos server and synchronised clocks, and can support SSO to other 'kerberised' services. It provides mutual authentication, and many-to-many communications.

Mutual Authentication – This requires that both the service provider and the user positively identify each other. In this way the authentication is strengthened for both parties; it cannot be phished or spoofed as users aren't tricked into entering personal information on fake sites.

Non-repudiation – The ability through historical logs and logical analysis to prevent or discourage an Entity from denying that it had acted as an Identity in a given transaction, especially in a legal sense.

Password – A credential, something only you know and that the authenticator can confirm.

Policy – A set of Rules, usually associated with a Role or other dynamic attributes. It is normally used for access provisioning and access reconciliation.

Privacy – a right to control the dissemination of the attributes of an entity. Attributes can be given up, after which it is dificult to restrict their use in the absence of any specific legal remedy.

Pseudonym – A fictitious identity that an Entity creates for itself, whereby the Entity can remain pseudonymous, or prehaps even fully anonymous, in certain contexts. Literally means "false name".

Realm or Domain - A realm or domain represents a single unit of security administration or trust.

Re-authentication – The same authentication is resubmitted by the known Identity, in order to “commit” a transaction that has been fully prepared during the session under the same assurance strength (this is deemed to be further protection from “session hijacking”). Alternately, the re-authentication may be required to ‘step up’ the assurance strength, so as to enact a transaction that requires higher security (such as two or three factors).

Registration – The process of an entity (re)establishing an Identity with a service provider.

Relying Party – The entity that relies on the result of an authentication. Usually, but not always, the same as the authenticating party and service provider.

Repository – A digital store, usually an LDAP directory or a relational database.

Risk – A measure of the (potential) impact that may be caused by the failure of an activity.

Role – The dynamic or logical associations, privileges or capabilities applying to multiple Identities, based on a set of one or more current Attributes. A role may have multiple identities, and an identity may have multiple roles.

Rule – The implementation of a decision that determines the Permissions of a Group, a Role or an identity whose access is based on particular attributes.

Security Token - A security token represents a collection of claims.

Security Token Service (STS) - A security token service is a Web service that issues security tokens (see WS-Security and WS-Trust). That is, it makes assertions based on evidence that it trusts, to whoever trusts it. To communicate trust, a service requires proof, such as a security token or set of security tokens, and issues a security token with its own trust statement (note that for some security token formats this can just be a re-issuance or co-signature). This forms the basis of trust brokering.

Session – A single Identity authentication period and its associated activity (temporary, non-persistent). Usually from logon until logoff or time-out, sustained with a local session cookie.

Signature - A signature is a value computed with a cryptographic algorithm and bound to data in such a way that intended recipients of the data can use the signature to verify that the data has not been altered since it was signed by the signer.

Single Sign On (SSO) - Single Sign On is an optimization of the authentication sequence to remove the burden of repeating actions placed on the end user. To facilitate SSO, an element called an Identity Provider can act as a proxy on a user's behalf to provide evidence of authentication events to 3rd parties requesting information about the user. These Identity Providers are trusted 3rd parties and need to be trusted both by the user (to maintain the user's identity information as the loss of this information can result in the compromise of the users identity) and the Web services which may grant access to valuable resources and information based upon the integrity of the identity information provided by the IP.

Token – A thing, a device, a physical item or software used to store attributes and credentials.

Trust (1) – an instance of a relationship between two or more entities, in which an entity assumes that another entity will act as authorised/expected.

Trust (2) - Trust is the characteristic that one entity is willing to rely upon a second entity to execute a set of actions and/or to make set of assertions about a set of subjects and/or scopes.

Trusted Source – A repository of identity information that can be relied upon for its accuracy due to the processes and security surrounding its creation and maintenance.

User – An Identity where the identifier of the identity is the public part of a paired Identity assertion. A user may have several identities / usernames / user-ids / logon-ids / sign-ons.

Verification – The process of confirming a claimed Identity.

Web Service – A standard means of web-based application to application communication, running on a variety of platforms and/or frameworks, sharing metadata, often publicly available to calling software.