An Investigation of Issues of Privacy, Anonymity and Multi-Factor Authentication in an Open Environment

Submitted in fulfilment
of the requirements of the degree
Master of Science
of Rhodes University

By
Shaun Miles

December 2007

Abstract

This thesis performs an investigation into issues concerning the broad area of Identity and Access Management, with a focus on open environments. Through literature research the issues of privacy, anonymity and access control are identified.

The issue of privacy is an inherent problem due to the nature of the digital network environment. Information can be duplicated and modified regardless of the wishes and intentions of the owner of that information unless proper measures are taken to secure the environment. Once information is published or divulged on the network, there is very little way of controlling the subsequent usage of that information. To address this issue a model for privacy is presented that follows the *user centric* paradigm of meta-identity.

The lack of anonymity, where security measures can be thwarted through the observation of the environment, is a concern for users and systems. By an attacker observing the communication channel and monitoring the interactions between users and systems over a long enough period of time, it is possible to infer knowledge about the users and systems. This knowledge is used to build an identity profile of potential victims to be used in subsequent attacks. To address the problem, mechanisms for providing an acceptable level of anonymity while maintaining adequate accountability (from a legal standpoint) are explored.

In terms of access control, the inherent weakness of single factor authentication mechanisms is discussed. The typical mechanism is the user-name and password pair, which provides a single point of failure. By increasing the factors used in authentication, the amount of work required to compromise the system increases non-linearly. Within an open network, several aspects hinder wide scale adoption and use of multi-factor authentication schemes, such as token management and the impact on usability. The framework is developed from a Utopian point of view, with the aim of being applicable to many situations as opposed to a single specific domain. The framework incorporates multi-factor authentication over multiple paths using mobile phones and GSM networks, and explores the usefulness of such an approach.

The models are in turn analysed, providing a discussion into the assumptions made and the problems faced by each model.

Acknowledgements

In completing this thesis I would like to thank my supervisor, Mr. Barry Irwin, for providing me with inspiration and the direction needed for such an undertaking. I would also like to thank my parents who have been patient enough to support me though my several years of being a student. It is finally time to become a productive member of society. A final acknowledgement goes to my fellow Masters students, who have put up with the endless rantings and ravings that were common in the labs.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Security is an integral aspect of any system. A system is defined by the boundary and interactions between the different components [122]. Security is a blanket term for ensuring the correct operation of the components, the interactions and the system as a whole [21]. In an open networked environment, such as the Internet, security is a means of protecting digital information and resources contained in systems. The Internet is a global network of inter-connected networks that represent a shared cyberspace with *ad hoc* decentralised and distributed control through the use of standard communication protocols. Within this space there are a myriad of individual components, whose behaviour is typically described in terms of interfaces for interactions.

As networking technology evolves and develops in response to the environment, greater abstract thoughts and concepts are built upon. With the advent and subsequent proliferation of the Internet, the traditional approach to security in the networked environment has become obsolete. As a result, there is a need for new approaches to be developed to achieve network security. Control and administration of individual networks and domains has to evolve to incorporate global connectivity. It is obvious that there is greater value for networks when there is unfettered and uninhibited information communication, such as cross-boundary inter-organisational collaboration and information flow. A new school of thought has arisen, in the form of De-Perimeterisation (De-P) [17], that challenges the traditional mindset towards security by attempting to redesign the security architecture at the ICT level to facilitate an open network that is inherently secure. It is a concept that is still in its infancy and looks to create a framework through the use of open standards. Within this framework, it will be a set of solutions that provide a defence in depth approach without the usage of typical security devices. Each solution should be characterised by being open, interoperable and Operating System agnostic. This keeps in line with global trends towards individual-centric security, greater collaboration between companies, the changing nature of the Internet, and infrastructure fragmentation.

A facet of this is Identity and Access Management (IAM) systems, which forms a conceptual connection between end users, network and domain administration, and the resources that need to be protected. As such there are disparaging needs, often that conflict, where the conflict is mitigated by compromise or a series of trade-offs. Security is typically a trade-off of usability, where greater security of a system negatively impacts the usability of that system. IAMs have value when constructed from open standards and protocols, as opposed to proprietary standards that exclude community support.

Typically, the definition of an IAM depends greatly on the point of view or the perspective of the party providing the definition. Gartner describes IAM as *"a means of finding an efficient, manageable, audit-able and secure way of connecting users or processes to enterprise resources"* [119]. IAM refers to *"those technologies that allow companies to manage and control user accounts and privileges and to enforce real-time access to resources"* [70]. It is also known by other names: Identity Management Architecture is *"a coherent set of standards, policies, certifications and management activities, for a specific business goals and objectives, with capability to evolve to meet future goals and objectives*" [122]. Another term is Authentication and Authorisation Infrastructures (AAI) [77], which is defined in terms of achieving an inter-domain authentication and authorisation service. This view point acknowledges the *ad hoc* nature of the open environment and aims to *"extend the scope of security solutions by providing an integrated authentication and authorisation service for communicating peers"*.

Essentially, an IAM is there to control access to protected resources in such a way that is secure and aligned with organisational strategic goals. An IAM has several goals and functions, such as identifying the entities within the system (system resources, users and the relationships between entities) so that authentication and authorisation may take place.

1.1 Chapter Overview

The basis for this thesis is defined and described in the problem statement in Section 1.2. The background of this thesis is covered in Section 1.3. This entails the move towards open environments and the advent of De-P, as it puts the problem statement into context.

Section 1.4 provides a summary of this chapter and an overview of the chapters comprising the rest of the thesis is in Section 1.5.

1.2 Problem Statement

This thesis performs an investigation into issues concerning Identity and Access Management solutions, with a focus on open environments. In the course of the investigation several subproblems are identified and addressed.

Firstly, the issue of privacy is an inherent problem due to the nature of the digital network environment. Privacy entails the user account representation and management, such that the disclosure and usage is a function of control of the owner of the information. Information can be duplicated and modified regardless of the wishes and intentions of the owner of that information unless proper measures are taken to secure the environment. Once information is published or divulged on the network, there is very little way of controlling the subsequent usage of that information. In reality, each IAM requires a different particular subset of user related information for the creation of an account. Should a user have accounts with several IAMs, the user would have to maintain several subsets of different credentials. IAMs have privacy policies that stipulate what the information is required for and how it will be used. This is a concern to users, where users may have little or no control over that information.

Secondly, the issue of the lack of anonymity is a concern for users and systems within a networked environment, in that the security measures can be thwarted through the observation of the environment. By an attacker observing the communication channel and monitoring the interactions between users and systems over a long enough period of time, it is possible to infer knowledge about the users and systems. This knowledge is used to build an identity profile of potential victims to be used in subsequent attacks. To address the problem, mechanisms for providing an acceptable level of anonymity while maintaining adequate (from a legal standpoint) accountability are explored.

Thirdly, in terms of access control, the inherent weakness of single factor authentication mechanisms is addressed. The typical mechanism is the user-name and password pair, which provides a single point of failure. By increasing the factors used in authentication, the amount of work required to compromise the system increases non-linearly. Within an open network, several aspects hinder wide scale adoption and use of multi-factor authentication schemes, such as token management and the impact on usability. This problem is examined with the idea of achieving a flexible and extensible framework incorporating multi-factor authentication.

The identification of the above mentioned issues results in a proposed generic framework being derived using open systems to solve the pervasive multi-factor authentication problem. This framework is then critically evaluated and discussed. Further contributions attempt to abstractly model approaches to solutions that exist already for some of the issues.

1.3 Background

This section begins with a brief look at traditional approaches to access control within closed systems, where communication complexity is reduced, in Section 1.3.1. Current developments in IAMs are still based on these concepts, forming the foundation for current advancements in the field. However, with the advent of the open network, IAMs require a new way of thinking, as discussed in Section 1.3.2. This need is reflected in the Jericho Forum, and the De-Perimeterisation approach, covered in Section 1.3.3. This explores the short comings of the traditional model, and the new possibilities that arise from open public networks. Section 1.3.4 identifies the need for this new way of thinking. Current trends in the development of IAMs are subsequently summarised in Section 1.3.5.

1.3.1 Traditional Approaches

Traditional approaches to authentication and access control are, as summarised in [77]:

- Discretionary access control - DAC
- Mandatory access control - MAC
- Role-based access control - RBAC
- Others
  - Clark and Wilson Model
  - Chinese Wall Policy
  - Personal Knowledge Approach

DAC models sets access rights (i.e. read, write, execute) between objects (i.e. resources being protected) and subjects (i.e. users and systems). Access rules are represented in an access control matrix. This kind of model is used in most operating systems. MAC models pertain to the flow of the information within the system, rather than direct ownership of objects and delegation of rights. Security classes are used to represent objects and subjects, where objects are labelled by a classification and subjects labelled with a clearance. This model creates multilevel systems, as content may appear differently to different levels of clearance. Within both models, DAC and MAC, policies surrounding linking users with permissions and resources are complex and prone to errors. RBAC models concern roles describing organisational functions and the minimum resources required to perform that role. This follows the principle of least privilege. Here two types of associations are maintained, between users and roles, and between roles and permissions. From this, permissions are authorised for roles and roles are authorised for users. The other models mentioned are not as pervasive as the previous models. The traditional approach models work well in homogeneous environments, but is ill suited to a distributed or networked environment [77].

As computing has become more ubiquitous, and with the proliferation of the Internet, there has been a re-evaluation of the definition and understanding of what access control is. Identity and Access Management has grown out of that re-evaluation, there to describe the new way in which the open networked environment is viewed.

1.3.2 Towards Open Environments

It has been acknowledged in an article [108] that there exists a gradual erosion of the hard-shell network perimeter that has been the mainstay of traditional corporate networks. Through web applications, email, the growing number of mobile and remote users connecting to the corporate network, the network perimeter is becoming more permeable. As a result, the network is becoming more difficult and costly to secure following traditional security principles laid down before the Internet became pervasive. A new way of thinking about security is required to cope with the myriad of malicious activity and threats that abound the Internet. De-Perimeterisation (De-P) is a concept that is challenging the current approach to network security. De-P is achieved by redesigning the security architecture at the ICT level to facilitate an open network that is inherently secure. It is a concept that is still in its infancy and looks to create a framework through the use of open standards. Within this framework, it will be a set of solutions that provide a defence in depth approach without the usage of typical security devices. Each solution should be characterised by being open, interoperable and Operating System agnostic. This keeps in line with global trends towards individual-centric security, greater collaboration between companies, the changing nature of the Internet, and infrastructure fragmentation. By examining the current literature surrounding the De-P concept, a definition that is both appropriate and relevant is achieved. Identifying the need for and the limitations of De-Perimeterisation sets the context for arguments for and against the move towards De-P. Highlighting the areas relevant to achieving De-P will help motivate further research in those areas, promoting their importance. Finally, addressing current technology and *"Best Practices"* techniques that represent or can be used in the first steps towards De-P will further motivate further research into the subject.

De-Perimeterisation is a buzzword that has been bandied about the industry since its inception in 2004. The following section seeks to clarify the definition within the Jericho Forum context.

1.3.3 De-Perimeterisation

The Jericho Forum put forward a broad definition for De-Perimeterisation in a visioning white paper [17],

"The act of applying organisational and technical design changes to enable collaboration and commerce beyond the constraints of existing perimeters, through cross-organisational processes, services, security standards and assurance".

De-Perimeterisation (commonly known as De-P) is a definition for a new security architecture and design approach. It is a concept centred on the creation of standards that will ensure secure communication across open networks. This will be achieved by the re-appraisal of the current ubiquitous traditional security model, known as the hard-shell or hard-perimeter model. This involves the gradual elimination of current security controls that exist at the boundary of the organisational network and the Internet, and by placing refined controls at the data and system level. The achievement of these goals has been road-mapped between five and eight years, where there will be phases for the gradual development and implementation of standards that encapsulate and promote the concept of De-P.

It is more than a concept; it is a way of thinking about security. Current mindsets towards traditional security approaches are a barrier to change. The Jericho Visioning White Paper [17] notes that this is because of fundamental traditional assumptions about ICT infrastructures. Firstly, that the organisation owns, controls and is accountable for the ICT infrastructure it employs. Secondly, organisations assume that all individuals sit within the organisation, not taking into account the increase in globalisation. This approach fails when dispersion, individual accountability, interconnection and fragmented ownership of infrastructure, accountability and access rights are additional requirements of the organisation's infrastructure.

Drive Behind the Concept

It is a concept first introduced into mainstream industry by the Jericho Forum. The Jericho Forum is made up by a group of international IT customer and vendor organisations whose purpose is the development of open standards to enable secure and boundary-less information flows across organisations [43]. The Jericho Forum is hosted by the Open Group, a like-minded consortium of companies and organisations that span the various sectors of the IT industry [49]. The drive behind this consortium is to deliver products and services to the industry that will create an environment that is conducive to boundary-less information flow. This is through creating services and open standards to assure global interoperability of different products across different platforms, within and between enterprises. The point to take from this is that the drive is coming from Users (companies that are not vendors but make use of vendor solutions), and are trying to create a business case for a shift in security thinking by defining their own requirements. Communicating these requirements to vendors will help the vendors create products and solutions that are more in-line with the organisation's needs. This supports the point that companies, organisations and individuals have recognised a need to re-address their thinking about security in general.

Set of Solutions

The realisation of a De-Perimeterised world, a Jericho world, will be through a set of solutions within the De-P context that are interchangeable and interoperable, based on open standards [17]. For example, a corporate network for a bank will need greater levels of security than a network for an education institute. While both would be adhering to the security standards inherent to the De-Perimeterisation framework, the corporate network would use a greater subset of solutions than the organisation's network. The solutions should also be Operating System (OS) agnostic in order for the De-P world to be vendor independent. Interoperability then becomes an important feature in the De-P world solutions. An interchangeable solution implies that it is interoperable by virtue of its ability to be swapped for another solution. Being OS agnostic requires the solution to be interoperable with the same solution on a different OS, in the same way that Java [84] is compiled independently of the underlying hardware platform into byte-code which is used by the OS's Java Virtual Machine to interpret at run-time. Interoperability cannot be achieved, while satisfying the other requirements, without the use of open standards.

Standards

A standard represents a clearly defined scope or context in which to achieve communication between different hardware and software platforms [69]. Standards come in four flavours; the first being the proprietary standard which is owned and controlled by a single individual or organisation [51]. This is restrictive in the sense that the owner closely guards the standard and insists that all users buy into the vendor-specific products at the expense of not using any other vendor's products. Companies or organisations spend time and effort on developing and inventing certain products to leverage market advantage. The creation of a proprietary-specific standard arises from the need to communicate with the company's other products. The results of those efforts are labelled "trade secrets", and are such because if the companies shared this knowledge with their competitors, they would lose their competitive edge. The second and third types are the *de facto* and *de jure* standards respectively. Both terms come from Latin, *de jure* means "by right" or "legally", and is used to describe a widely used standard that is endorsed by a standards body [121]. *De facto*, on the other hand, means "in fact but not in law", and describes a widely used standard that is not endorsed by a standards body [121]. The fourth type is the open standard which creates an environment conducive to open collaboration and communication [69]. If one were to consider the Internet, its success is partly due to the globally accepted open standards that the Internet uses [106].

Open standards are achieved through a steering organisation (for example, the Internet Engineering Steering Group - IESG) chartering a working group (such as the Internet Protocol version 6 working group - ipv6) to develop and promote them [42]. The steering organisation and working group can be seen as a Standards Setting Organisation (SSO), as described by Krechmer [69], and follows one perspective of the term "open standard". In other words, the SSO creates an open standard if it follows the tenets of open meeting, consensus, and due process. This means that any party or individual can participate in the standards development process, where all interests are discussed and agreed upon with no domination by any group, and that a balloting and appeals system is used to find resolution. The other two perspectives of the "open standards" term, given that the standard exists, are those of the implementer and the user of the standard respectively [69]. The group behind De-Perimeterisation, Jericho Forum, advocates the creation and use of open standards, in order to develop truly interoperable De-P solutions [17]. Its development should be a collaborative activity as to maximise the output from the various stakeholders. The Jericho Forum is a SSO responsible for several work groups tasked within areas relevant to De-P.

Defence in Depth

Defence in depth refers to a security architecture that relies on the intelligent application of various technologies and "Best Practices" techniques [96]. It means having several layers of security that make the core infrastructure (the most valuable assets) more difficult to compromise, without relying on a single point of defence [68]. The measures taken are prescribed by top security specialists, referring to the best methods to secure several areas within the network and the end user. The Jericho Visioning White Paper [17] addresses defence in depth as being an important part in achieving De-Perimeterisation. Security is seen as a process rather than a product, and there are several integral relationships between defence in depth and De-P. Joel Snyder in a white paper [105] lists six defence in depth strategies that corporate networks can employ to provide additional layering of security. This architecture requires networks to be aware and self-protective. This implies a certain amount of autonomy in the administration of the network. Cisco is providing intelligent switches and routers that can, through favourable policies, quarantine machines that are not up to date with software patches, or machines that have been found to be compromised [72]. Snyder [105] puts forth six strategies that, when all six have been deployed, significantly change the security setup of the network. These are:

- Authenticate and authorise all network users
- Deploy VLAN's for traffic separation and course-grained security
- Stateful firewall technology at the port level for fine-grained security
- Place encryption throughout the network to ensure privacy
- Detect threats to the integrity of the network and re-mediate them
- Include end-point security in policy-based enforcement

This is aligned with the De-Perimeterisation emphasis of protecting the data through encryption, and shrinking the security bubble [88]. The bubble symbolises the boundary between the "secure" inside and "insecure" outside. By decreasing the traditional perimeter boundary to smaller security islands, bubbles that protect separate network entities, the overall risk to these islands decrease. This is because if one island were to be compromised the other islands would still be relatively secure, depending on the controls and policies in place.

1.3.4 Identifying the Need for De-Perimeterisation

The Erosion of the Perimeter

The erosion of the hard-shell perimeter has been acknowledged by security professionals since the proliferation in use of the Internet by the corporate network [108]. Mobile computing and remote connections add to the vastness of domains that are not under the immediate control of the corporate network. As remote connections to the corporate network from public networks increase, the gap between the different trust domains increases. Mobile devices with wireless capabilities present an intermediate launching platform for attacks on the corporate network as they are more difficult to secure. The act of tunnelling, wrapping a protocol within another protocol, can bypass stateful firewalls and provide an uninterrupted passage into the network [87]. Web applications and email are prime examples, as this requires content filtering, generally performed on the end-point machine. The Jericho Forum identifies the trends that support the gradual erosion of the hard-shell network perimeter [17]. There is a move towards centralising ICT assets (from distributed branch-specific servers and data centres) into fully redundant data centres [88]. This means that the traditional hard-shell is fragmenting and there is greater interconnectivity of corporate branch networks. Another trend is the increase in need or use of open networks (The Internet).

Increase in the Openness of Networks

There is an increase in corporations collaborating with each other, and in using the Business Process Outsourcing (BPO) industry sector, requiring communication between entities to maintain confidentiality, integrity and availability [5]. The use of BPO means that corporations can outsource IT requirements to specialists in the field without having to deploy their own services. As noted in the Palmer paper [88], collaboration increases the number of stakeholders in the organisation's infrastructure. Managing the infrastructure then becomes difficult for a single entity to carry out, as the number of connections and the distributive nature of systems increases complexity. In a white paper from Open Groups [56], it describes a drive to make boundaries more permeable to achieve better integration of the economic and functional business processes of an organisation. De-P is built upon this principle. The openness, or permeability, of networks is hindered by four types of boundaries [17]:

Infrastructural: Inhibiting interconnection, and lacking the underlying facilities to interconnect

Structural: System growth is limited by the scalability of its structure

Architectural: Different architected technologies are sometimes not interoperable

Semantic: Different ways of representing the same things and are difficult to reconcile

Open communication is desirable in an ever increasing e-commerce market, as more business opportunities arise through collaboration. Organisations overcoming boundaries of open communication will benefit from increased economic activity, and the streamlining of current business processes.

Return on Investments

Return on Investment (ROI) is essentially cost management, as pointed out by a Cisco white paper [114]. The cost of deploying a security-related solution is contrasted with the risk the solution is to mitigate [87]. In a profit driven market, cost versus risk analysis dictates what security decisions are implemented. As networks become more fragmented and trends towards smaller security bubbles increase, the cost of securing the boundary of the network increases. More hardware devices (firewalls, IDS, IPS), and other controls, will be needed as more avenues into the network boundary are opened up. The traditional security architecture is not well suited to scalability, since it relied on static boundaries [56]. Mobile computing, remote connections and distributed networks make the border more dynamic. In response to vulnerabilities and other new avenues of attack, security is tacked on to deal with these new threats. Since systems are not homogeneous, one cannot be certain that security solutions deployed will be interoperable. As the complexity of an infrastructure increases, vulnerabilities might be overlooked, causing more problems in the long run. There are more long term costs attached to this approach than short term benefits. Identified in the paper [114], within a cost savings and avoidance context, a common basis for security throughout the infrastructure creates benefits in management, visibility, control and enforcement. That is, if a standard were to be created, there would be a convergence of disparaging systems and their security efforts that would decrease complexity and increase interoperability. A greater understanding of the infrastructure's security state allows greater predictability in the scope and depth of proposed changes in security posture. There is a need for the convergence of security solutions, as to enable them to communicate effectively so that the enforcement of security policies is more effective. Consider web services, since its inception, the definition of web services has been evolving, and so has its benefits. As stated in [120], web services provide simplified mechanisms to connect applications regardless of technology or platforms. Through the use of industry standard protocols with universal vendor support provide low cost communications and business integration. The benefits of following the web services include IT cost and complexity reduction, business wide process streamlining, and business consolidation. The hindrance web services have faced, as noted by a Gartner article [6], has been the lack of standards-based mechanisms for ensuring quality of service across the Internet. This has led to poorly implemented web services. Another problem is that standards lacking the participation of some key vendors has led to interoperability issues throughout web services deployment.

Individual-centric Security

In the Palmer paper [88], the author identified that there is a trend towards individual-centric security. Concisely, this means the securing of access and controlling the use of information within the perimeter. This has already been identified as a key strategy for defence in depth [52], where access privileges depend on policy enforcement, such as the machine users log in on and the role of the user. Aspects such as compliance, whether or not the machine is up to date with security software and the latest patches, trust domains and security bubbles, authentication and Identity Management will become important features in the enforcement of end-point security policies [88]. Once again, this issue will be confounded by complexity in building and maintaining the business logic of dealing with different platforms and security postures. The complexity of managing different states can be alleviated by following a standard that encompasses these aspects, supporting the move towards De-P.

1.3.5 Current Trends

In a 2006 survey of 1708 U.S tertiary education institutes, conducted by the EDUCAUSE Current Issues Committee [33], Security and Identity Management was reported as the number one "most important" IT-related issue among the 628 institutes that responded. The survey was directed at the IT administration departments of the education institutes, with the purpose of extracting the most pressing issues that needed to be resolved for the strategic success of that institute. Security and Identity Management, incorporating both security and IAM aspects, is defined as the balance between expanding information access and the requirements for providing protection from unauthorised access and abuse. The reason for this becoming a high priority issue is the general move towards the digitisation of information and resources as an increasing number of education and administrative activities are carried out in a networked environment. Several sub-issues specific to IAM were identified:

- Balancing the needs for security and impact on usability that increased security incurs
    - Multi-factor authentication schemes
    - Information security training and awareness
- Information and resource security classification
    - Specific controls and usage policies for each category
    - Managing risk of identity theft and privacy issues
- Security policies
    - Up-to-date and enforceable, implying a regular security policy review process
    - Reflect the organisational priorities and strategic goals
    - Policies and technologies for external communication and partnership collaboration
- Digital identity management strategies
    - How identity information is represented, stored and managed
    - The usage of standards and handling of non-compliant systems
    - Ownership of identifying data stored on the system
    - Regulatory compliance and information laws and acts
- Technologies
    - Support strategic goals of organisation
    - Incorporating standards to increase interoperability
    - Adaptive security processes to patch and update critical areas
    - Controls and monitoring of security activities

Although the survey was directed at academic institutions and organisations, the sub-issues presented above have direct relevance for business oriented organisations and their networks. Rather, these issues should be considered for any organisational network. Consider the first issue related to balancing the needs for security and the impact on usability, where increased security measures tend to hinder usability. For example, a bank has several layers of security protecting the vault that restrict employees from freely accessing it, rather requiring them to follow predefined procedures. Increasing security controls, or making assertions about security activities limits the freedom of the operators and users by increasing the effort on their parts in making use of the protected resources and systems in terms of usability. However, this feature of security can be mitigated by adequate training and awareness, enabling the users to come to grips with the technology and controls in place, to become more at ease in their use as well as the understanding as to why such controls are in place.

This has direct relations with another identified sub-issue, where the effort spent on protection of resource should be related to value and the risk of compromise. This implies a means to classify and categorise resources and systems in such a way that there is an efficiency in the defence of a network, that separate controls do not conflict with each other and provide holes through which to attack. Being able to identify resources that are important to the organisation and its members, the ramification of that resource being destroyed, modified or stolen, and the probability of any such event occurring is an integral part in Risk Management [109]. Risk Management has direct application in understanding Identity Management. This involves assessing the threats that can exist for each identified component of an IAM and creating controls that mitigate that risk. The categorisation and classification of resources and systems means that everything in that set is identified and can have specific controls and policies attached to them that determine *who* can access *what* and in what manner (*how*) it may be used.

Policies represent varying levels of intelligence, from simple rule-based conditions to expressive subsystems that define how individual components can interact. Essentially, policies define a wide range of activities, from low-level access to statements of intent (mission statements). In terms of security, policies are used to define areas of operation using criteria and conditions that are evaluated to determine a course of action. As identified in the survey, the correspondence between organisational priorities and strategic goals with the actions a system should take in a given situation is important and should be correctly captured and expressed in policies. A policy is meaningless unless its decisions are enforced. For instance, when an organisation expresses in its privacy policy that any personal user information stored in its systems shall not be disclosed to third parties, the organisation should have security controls in place that limits access to such information only to people with the proper authority and understanding.

The next sub-issue encapsulates the definition and expression of strategic goals of the organisation's IAM system. These goals entail how identity information is represented, stored, and managed, taking into account notions such as the ownership of the information. The question of who owns the information determines what can be done with that information. For instance, if an organisation owned and had control over the stored identity information then that organisation could utilise that information in a way that would further its strategic goals. This extends to the notion of compliance, that is, information Regulatory Laws and Acts, ensuring that systems and information privacy conform to specified controls. This depends on the type of organisation and the nature of the information, where regulations and laws are enacted to protect the rights on both the organisations and the public who make use of them [40, 97]. An essential means for the organisation to ensure compliance with a regulatory body is to make use of open standards. A standard represents a clearly defined scope or context in which to achieve communication between different hardware and software platforms [69], where an open standard creates an environment conducive to open collaboration and communication. Non-standard systems are less compliant as their design and implementation are done without an established template and method for achieving goals, which makes it more difficult to measure compliance.

The final sub-issue ties all the previously mentioned issues together, which is the identification and application of technologies that address each issue. Specifically, technologies that will help and that are integral to the achievement of the organisation's strategic goals in terms of security and IAM. A report by the research company Gartner [70], identifies and comments on the state of IAM-related technologies, illustrated as a hype-cycle in Figure 1.1. By way of explaining the information presented in the diagram, the definitions of the different stages in the cycle are presented below.

Technology Trigger An event that generates significant industry and press interest.

Peak of Inflated Expectations As the technology is explored with over-enthusiasm and unrealistic projections, more failures than successes are recorded as the technology is pushed to its limits. This is a learning period where results are presented in conferences and publications.

Trough of Disillusionment With mounting failures and the wane in media and public interest, the technology becomes less of a priority.

Slope of Enlightenment Subsequent experimentation and development allows for the understanding of the technology in terms of its applicability, risks and benefits. The development process is aided by the creation of commercial tools and methodologies.

Plateau of Productivity The tools and methodologies have been through several generations, with the technology's real-world applicability and benefits solidly demonstrated and accepted. Approximately 20% of the target audience are adopting or have already adopted the technology.

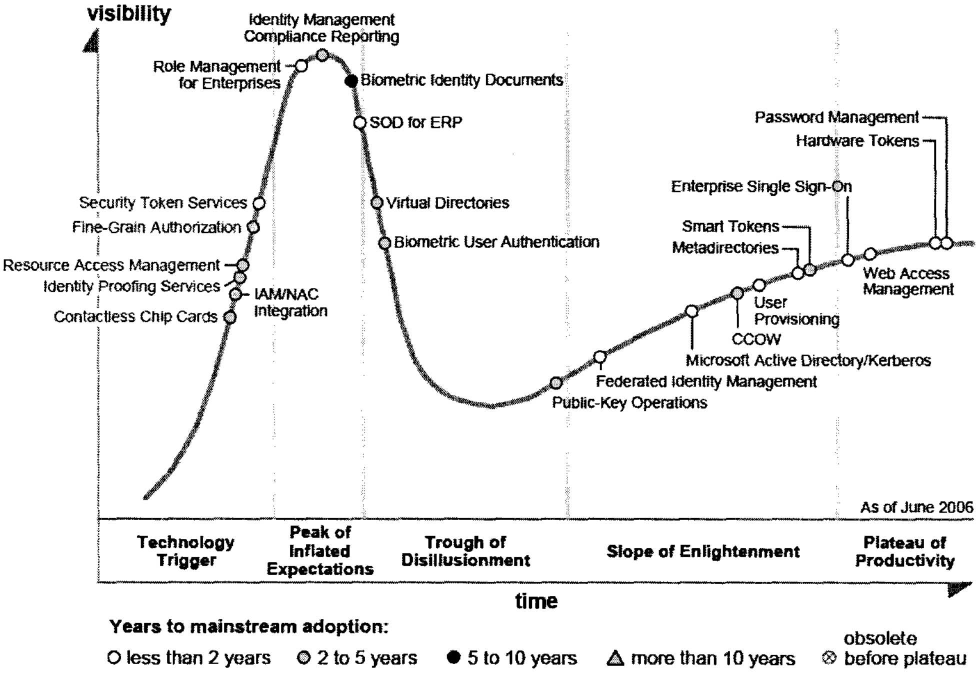

Figure 1.1: Gartner Hype Cycle for Identity and Access Management Technologies for 2006 [70]

The diagram depicts the technologies that are being recognised by industry as viable solutions to real problem domains within the realm of Identity and Access Management. It also indicates the prediction of when these individual technologies will become widely accepted.

1.4 Chapter Summary

In this chapter the problem statement summarises the goals and aims of the thesis. This is further contextualised by delving into the background for engaging in this thesis. This covers the movement from the traditional approaches to securing a network, which became obsolete with the proliferation of the Internet, towards an open environment. With this a new approach, embodied in the design paradigm of De-P, the basis for this thesis is set. This is further supported by discussing the current trends within the field of IAM.

1.5 Thesis Overview

A literature review covering the concepts relevant to IAM, such as identity and access control, are provided in Chapter 2. This includes trust and management and a brief overview of policies. Chapter 3 delves into the issues stated in the problem statement, namely privacy, anonymity and multi-factor authentication in open environments. This is achieved by further literature research.

The issues of anonymity and privacy are addressed in separate models in Chapter 4. Each model attempts to describe, at a high level of abstraction, how its relevant issue can be addressed.

Following the identification of the lack of a pervasive and scalable physical security token, Chapter 5 provides a framework that incorporates the mobile phone as second factor of authentication.

An analysis and discussion of each of the models presented are performed in Chapter 6. This entails the assumptions made and the problems faced by each model. A discussion, through a series of scenarios that unite the models, in terms of the De-P approach is also provided.

The thesis is finally concluded in Chapter 7, re-addressing the problem statement with what has been achieved.

Chapter 2

Literature Review

2.1 Chapter Overview

In the previous chapter the background concerning IAMs was discussed, showing both the need for new ways of approaching open systems and the trends within industry. This chapter explores the fundamentals of IAMs, in that we take a look at the concepts and components that constitute an IAM. Discussed within is a broad cross-section of literature concentrating on two main sections: identity (Section 2.2) and access control (Section 2.3).

Within Section 2.2, the concepts that surround identity are covered, with a focus on federated identity in Section 2.2.1. Models of federated identity systems, used to describe different approaches, are presented in Section 2.2.3. Section 2.2.4 addresses the different approaches to providing identity stores, in that the means to store and make available identity information. Section 2.2.5 details another aspect of identity management, that of Public Key Infrastructures, including variants on this theme.

Section 2.3 has the task of describing access control related concepts. The access control feature covered is authentication and authorisation in Section 2.3.1. Section 2.3.2 attempts to deal with the open-ended nature of trust within a digital environment. This includes notions such as trust management (Section 2.3.3) and trust negotiation (Section 2.3.3). An inherent and critical aspect of access control and trust negotiation is that of policies and policy specifications, this is covered in Section 2.3.4.

Finally, the chapter summary is provided in Section 2.4.

2.2 Identity

Identity is information indicating the uniqueness of the object (a user or machine), distinguishing other objects from itself [26]. Identity, within a digital environment, is typically categorised and disclosed as *"something you know*"," *something you have*" or *"something you are*". In the real world, people within a small and local scale are recognisable at a social level through the myriad of social interactions that occur. Our identities are inherently linked to who we are as people within in society. On a national and international level, Governments issue identity documents that uniquely identify an individual based on several immutable criteria (such as name, date of birth, and country of citizenship). Through diplomatic relations are these government-issued identity documents accepted. In the digital world it is much more difficult to establish the true identity of another person, as well as establish a global identification scheme. Following the tenants of free speech, the Internet is not regulated for content (other than Intellectual Property infringements) at the moment and has grown from an *ad hoc* collection of networks. Users of the Internet hold high their right to privacy and anonymity in cases where it is possible. There are, of course, some aspects of Internet usage that require that users provide personal information in exchange for otherwise free services.

According to Windley [122], *"naming is one of the fundamental abstractions for dealing with complexity"*. As such, naming becomes the first way in which a person deals with communicating a description of an object, and is one of the most common attributes of identity. A name exists within a name-space, defining the universe in which that name has meaning. The uniqueness of names and as well as what the name references, depend on the properties of the name-space. This allows a decoupling between name and description. Take Internet Domains as an example, the domain name *ru.ac.za* might be bound to a certain IP address, allowing those that know the domain name find the machine, however, should the machine change or the IP address change, that change will be transparent to those that type in the domain name. Domain names also help distinguish between different services on the same machine. When considering identity, it is important to consider the name-space in which the identity belongs. Identity and Access Management is an industry sector that is still in development [17], which can be divided into three levels, each defining the limits of its own name-space:

Enterprise - an identity is under the control of a single enterprise and has no meaning outside that enterprise

Federated - an identity is linked to more than one enterprise by a common identity management system

Global - an identity represents a unique real-world person

Within the context of this thesis, Enterprise identity management is limited to a single domain, and thus is limited in functionality and usefulness. If one were to consider the case of a global identification scheme, global standards are tricky to enact since there has to be global agreement. Taking into account international relations, it is perhaps not feasible trying to get U.S.A. to interact with China at that level. Perhaps one application of a global identification scheme free from the afore-mentioned problems is grid computing [112]. Grid computing is characterised by the fact that computer devices, processes, memory and disk storage are shared over the entire grid, regardless of geographical location. Explored in Lopez *et al.* [78] where the difficulties inherent in creating and maintaining a global name-space are: who controls the name-space, and everyone has to agree to the operation of the global name-space. Furthermore, those in control of the name-space can have access to user information stored there, requiring people to submit their personal information to the control of others. A possible solution to this is to have domain-specific identities that are unique within that context, while the domains themselves are distributed and integrate on a peer-to-peer level. Thus control is localised to a specific domain, limiting the visibility of a user's credentials.

2.2.1 Federated Identity

Federated Identity is a model of distributed identity management [82] that sees wide-spread use in the realm of web services and browsers [92]. Federation, in the words of Windley [122], *"defines processes and supporting technology so that disparate identity stores can cooperatively solve identity tasks"*. This approach grew out of the need for collaborations, embracing cross-boundary information flows. This means allowing users from an enterprise to engage and use services offered and controlled by another enterprise, as well as opening up a new industry, Business Process Outsourcing (BPO). It also allows enterprises to share network administration and maintenance costs, where the enterprises engage in a symbiotic relationship. Browsers and web services leverage the benefits of platform-independent and mobile-based implementations, and closely follow the Jericho Forum thinking in terms of loosely-couple components. This requires the development of standards, the basis for wide-spread acceptance and usage.

Liberty Alliance

An example often used when describing federated identity is Liberty Alliance. The Liberty Alliance (LA) Project is based on SAML and WS-* specifications, and makes public proposals without an open standardisation process [91], meaning that input is accepted from a select group of people rather than the community or industry. Liberty Alliance, however, aims for an open, federated, single sign-on solution for a digital networked environment, spanning several domains, creating a *"Circle of Trust"* [77]. In this sense, open is intended to mean interoperability between domains, using various technologies, protocols and standards.

Within Liberty Alliance , there are Identity Providers (IdP) and Service Providers (SP), and Principles (users) belong to an IdP, where users authenticate themselves using credential disclosure, and resources are protected by the SP of which authorised users can access [22]. The Liberty Alliance specifications form frameworks that describe how entities can interact with each other, defining the *"Circle of Trust"*, the basis for cross-domain information flow [101]. The LA specifications mirror business relationships, allowing heterogeneous systems and domains to integrate across geographical and administrative differences [48].

2.2.2 Components of Federated Identity

This section defines the components that make up a Federated Identity Management system, as well as various models expressing their interaction. This is based on work by Djordjevic and Dimitrakos [34], where a federation system is an aggregation of the following component's functionality . It is important to note that the models presented below are an abstraction of a federation system's functionality, where actual implementations may have components performing several other functions.

STS - security token service

Issues tokens on a claim, validates the validity of the claim and validates the exchanges of tokens in a specific token format.

CPS - credential processing system

Transforms credentials between 'bound' and 'free', specifically credentials or security attributes that are recognised within the trust realm and those that that can be used outside the trust realm.

PDP - policy decision point

A network node that makes a decision based on agreements between identity service providers (represented in a policy). It is the minimum required amount of information to disclose for a decision to be made that is held in the policy.

PEP - policy enforcement point

Any mechanism that enforces a security policy, including message inspectors, interceptors/gateways, secure message routers and applications.

2.2.3 Models

These models represent the different manners in which the components can interact, each with their own particular advantages and disadvantages, as presented in [34]. The PEP is generally responsible for handling the incoming/outgoing message requests through a 'message context handler/dispatcher'. The models are the specific interaction of the PEP with the rest of the components. There is big divide between the benefits of decoupling components versus the overhead costs of interaction messages. Component interaction and the message sequence depend on the model and the policy implemented. Some policies may require multiple rounds of credential validation and authorisation evaluation before a final decision is made.

PEP-biased

The PEP facilitates interaction and controls the data flow, and is responsible for maintaining the messaging sequences. For incoming messages the STS is asked to validate and approve claims. The CPS provides further information of the identity of the claimant. The PDP then makes an authorisation decision that the PEP enforces. The CPS obtains the set of credentials for the request. The STS then issues a token to the requester. The advantage of the model is that the components are loosely coupled and is useful in large decentralised networks. The disadvantages are a large message overhead, and that an intelligent decision policy is required to be implemented.

PDP-biased

There is more emphasis placed on the processing of security information and the reasoning performed by the PDP. The PEP effectively only knows about PDP components since the token and credential validation is performed as part of the authorisation policy evaluation within the PDP. The PDP needs to implement the CPS/STS functionality and the handling of context messages. Incoming messages are handled by the PEP as it sends all evidence available to the PDP unprocessed. The response from the PDP indicates the authorisation decision. For outgoing messages the PEP sends an authorisation request to the PDP to obtain an authorisation response. The advantage is that validations of credentials and the evaluations of authorisations can occur concurrently due to the interactions of the PDP with the CPS/STS components. The disadvantage is the restriction of flexibility of the model, as any updates to the CPS/STS will need to be require the PDP to be updated. The components are the tightly coupled and thus result in system designs that are difficult to maintain and manage.

STS-biased

The STS needs to implement PDP/CPS functionality, and is similar to the previous model in that regard. For incoming messages the PEP requests the validation of tokens for a particular action. The PDP and CPS aid in the validation process. For outgoing messages the same occurs. Advantages include concurrent interactions to reduce delay. The model can allow deployments of multiple STS's. The policy and validation process can be decoupled. Disadvantages are the scalability and flexibility problems inherent in this model. There is an increasing dependency on specific types of tokens.

Hybrid-based

This models offers better management of STS and PDP capability aggregation while removing the overhead of the controlling the flow of enforcement actions. For incoming messages, the PEP asks the STS to validate tokens specific to the action requested. The PDP then returns the authorisation decision. For outgoing messages, the PEP requests an authorisation decision from the PDP to continue with the request. The STS will then issue a token to the PEP. This model improves the flexibility through decoupling the essential STS/PEP and PDP/PEP interactions. It suffers from the tight coupling between the CPS, STS and PDP. This tight coupling may introduce dependencies that complicate the federation of trust realms.

Broker-based

This model further decouples the context handler/ dispatcher mechanism in order for it to become a sort of broker between the other components. It manages the interactions and flow of messages between the components and enforces security actions for incoming/outgoing messages. This maximises the decoupling of the components, allowing greater flexibility, and also eases the interaction between the different components. This is achieved at an increased cost of overhead messages, reducing efficiency.

2.2.4 Identity Stores

Since identity information, representing individual and separate entities within a name-space, requires persistence for the system to make accurate access control and authorisation decision [122], the concept of an identity store must be addressed. An identity store can be viewed as a database or a directory, where a database is flat and relational, while a directory is usually hierarchical. According to Windley [122], a directory service is a network-aware directory that allows distributed applications to make use of centrally managed identity information. These range from RMIRegistry (a Java Remote Method Invocation facility) to the X.500 family of directory services and to LDAP (Lightweight Directory Access Protocol) implementations.

Single sign-on (SSO) has grown out of the need to aggregate disparaging identity sources to create a unified user experience, forgoing the complexity of multiple identities and passwords within a domain. There are four approaches:

- Build a single centralised identity store
- Create a meta directory that synchronises data from other identity stores in the organisation
- Create a virtual directory that integrates identity data into a single view
- Federate directories by bringing identity stores together (distributed)

Federated identity creates a single view of identity within a name-space by bringing information together from different sources.

Meta and Virtual Directories

Meta Directories are aggregated collections of directory information, which creates a single view of identity based on dynamic queries from different sources [122]. The identity information are stored in different systems and are transparently brought together as an abstraction of that data. This allows the underlying implementations and architectures to change while applications making use of the meta directory still point to that source, while the meta directory is responsible for handling the change while that application remains unaware. As such, the meta directory becomes the single point of administration, avoiding the need for accessing multiple interfaces to maintain the data. Redundant information can be eliminated in this approach. This contrasts the federated identity approach by allowing the different sources to be separate and be accessed only when required. There are, however, challenges in building a meta directory service; governance and implementation issues that have to be addressed. Governance includes issues such as ownership, inter-organisation relationships, law and privacy issues, and administration concern. Implementation issues include the architecture, protocols, data format and data synchronisation. Meta directories employ software agents that replicate and synchronise data to a single location.

Virtual directories aim for the same end as meta directories, though the approach is somewhat different. It still provides a single view of identity information, using real-time queries, mapping fields from the virtual schema to fields in the physical schema of the real directories. A query is made to the virtual directory where several separate queries are made in real-time to the connected physical directories, where the results are then aggregated before being returned to the application. This represents a real-time interface to multiple data stores, creating a standard view using a standard API.

Meta Identity

A meta system is defined as a system of systems, and works by tying separate identity systems together into a larger interoperable system, from the*"Laws of Identity"* as presented by Kim Cameron and Michael Jones of Microsoft in [20]. This differs from meta directories in that a meta system is *user centric*, as per the first law [19], where control and consent lie with the user. Meta and virtual directories are design approaches that favour the organisational perspective, and meta identity systems such as CardSpace is from the end-user perspective [20].

The aim of meta identity systems is to be able to integrate disparaging identity systems, representing different technologies and standards, such that user involvement is transparent. By allowing identity systems to remain separate and seamlessly integrating individual relying parties (entities that require identities to operate), it allows the user experience to be free from concerns of changes in the underlying technologies and standards.

The *"laws of identity"* define the architecture of a meta identity system [19]:

1. User control and consent
2. Minimal disclosure for a constrained use
3. Justifiable parties
4. Directed identity
5. Pluralism of operators and technologies
6. Human integration
7. Consistent experience across contexts

Law 1 and 2 directs the control and disclosure of user identity information, where the system requires the consent of the user in order to make use of identifying data, and that the system will disclose the least required set of credentials during transactions. Law 3 expresses that the credentials can be disclosed to parties with a necessary and justifiable relationship with the user. Law 4 contrasts between public (omnidirectional) and private (unidirectional) space identification, in that support must exist for both types in the system. Omnidirectional identifiers exist in a public space, where it is clear what an identifier is identifying. Unidirectional identifiers are known to a single party and exist in a private space, such that there is no correlation between the identifier and the object identified in the public space. Law 5 supports the interoperation of multiple identity technologies operated by multiple identity providers. Law 6 propounds the notion that humans are a distributed component of an identity system, and as such, should be integrated with unambiguous human-machine communication mechanisms that inherently protect against identity theft. Law 7 expounds the separation between the user experience and underlying technologies it makes use of. In such away, the user experience will be consistent regardless how the identity technologies change and evolve over time.

As such, meta identity systems represent an emergence of a new paradigm; *user centric* identity management systems, where the idea is to give control to the user over the use and disclosure of their own identity information [12]. This is a divergence from traditional and organisation views of IAMs, where the design focus is on the administrative side of operations; known as *provider centric*.

2.2.5 Public Key Infrastructures

Another means of identifying a user is through the use of Public Key Infrastructure (PKI) certificates. Traditionally, PKI models are centralised where a Certificating Authority (CA) issues a certificate binding an identity to a public key [102], as illustrated in Figure 2.1. The CA acts as a trusted third party that acts as an intermediary, allowing everyone to trust a single entity rather than to trust everyone else. This assures less work for the person who uses the CA since trusting a single entity is easier than trusting or distrusting many entities. This forms a hierarchical structure. However, PKIs and CAs have many problems: at the root of trust is the CA, a big question is who certifies the CA, which ends up being recursive. Another issue is that of the private key, when issued with or generating a public/private key pair, a user has to store the private key and keep it secure. Another problem is that of revocation, when a private key is compromised, an attacker can forge messages with the stolen key. Revocation is the act of determining that a key is compromised and communicates that to the environment or the CA through the use of Certificate Revocation Lists (CRL).

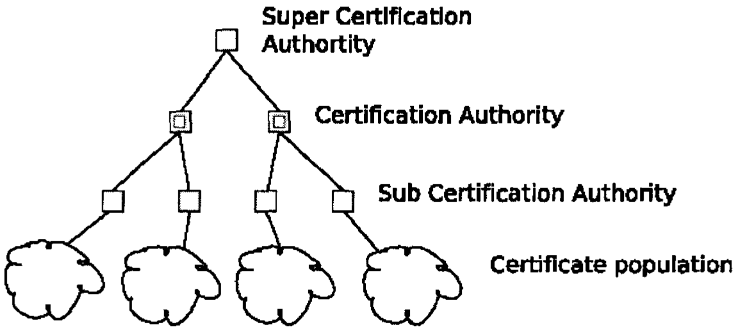

Figure 2.1: A model of a centralised PKI

X.509

X.509 is a Public Key Infrastructure Internet standard based on earlier developments of X.500 [16]. This is a pervasive standard in the Internet, as is it was developed to that end. The use of X.509 certificates extends to World Wide Web applications, electronic mail, user authentication and IPSec [57]. The Internet Engineering Task Force (IETF) X.509 working group (PKIX) have been tasked with the development of a specification, a standard, to evolve and adapt to the needs of the Internet. Like in PKI schemes, a X.509 certificate binds a Distinguished name to a cryptographic key, issued by a central authority. The X.509 certificate format is a standard for Public Key certificates, as widely used [24]. While X.509 certificates provide a means for strong authentication through the use of digital signatures, it lacked a sure means for providing equally strong authorisation control. Through later revisions (revision 4), a Privilege Management Infrastructure (PMI) mechanism was added, providing a standardised method for strong authorisation [24]. In short, sets of attributes are bound to a certificate (called an Attribute Certificate), forming the primary data structure in a PMI. These attributes are used to describe privileges afforded to the user (or holder of the certificate) as bestowed by the issuer. The issuer is called an Attribute Authority, where attribute certificates allow for a wide range of uses [9]. Here, X.509 attribute certificates are used to provide anonymity. Furthermore, it is possible to mirror traditional methods of authorisation, such as Discretionary Access Control (DAC) and Role Based Access Control (RBAC), as presented in [24]. Privileges can be used to represent roles, as well as the required attributes a user has to have to be able to fulfil such roles.

Decentralised PKI

Lopez *et al.* [78] states technical, economical, legal and social reasons why PKI has failed its goals. The more relevant issues are discussed under the technical reasons why PKI has failed. The complexity of the deployment and operation of the infrastructure, certificate management, and incorporating a global name-space are valid reasons not to implement a PKI. Instead, if a PKI were to be established that was reliable and robust, then outsourcing to this PKI would work well in an eCommerce setting. The work in this paper however, was limited to a centralised hierarchical model of PKIs.

Due to the shortcomings of centralised PKIs, approaches using decentralisation were developed. Aberer *et al.* [3] presented work within the context of customer-to-customer eCommerce. The decentralised infrastructures can be categorised into three subclasses: Web-of-trust, statistical and hybrids, referred to in Figure 2.2. In the web-of-trust model, peers rely on other peers to certify the public keys of a peer. This is based on graph theory where obtaining a trustworthy public key of a peer is a matter of finding a path of within the graph of acquaintances. Using a simple transitive rule which relies on trust of peers in extending the web of trust, where A trusts B then A will trust C if B trusts C. However, such an approach is unreliable as its strength is determined by its weakest link. There are other deficiencies, since the path is found by doing

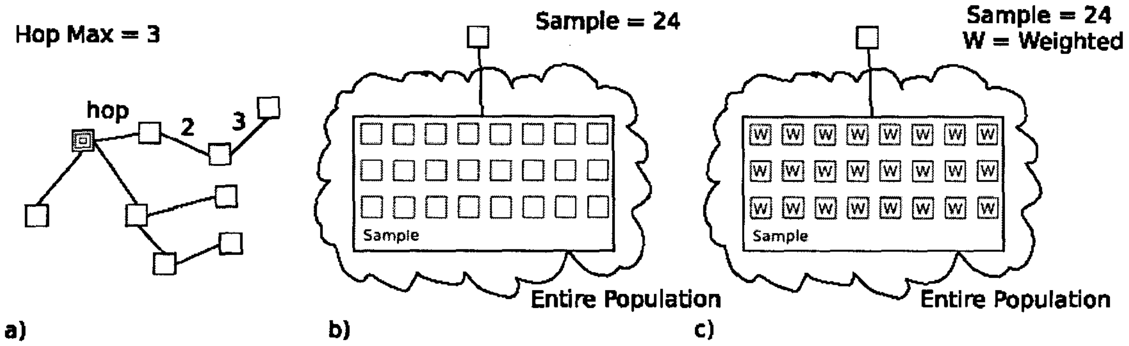

Figure 2.2: a) Web of Trust b) Quorum c) Hybrid Models for decentralised PKIs. Based on [3]

random walks through the trust graph. This approach is not efficient as the effort is not shared and has high, unbound latency, because in a structure-less peer-to-peer architecture searches have to be done by flooding the network with requests. It does not use the collective knowledge of the whole population, but rather a small subset which are derived by a limited number of transitive hops in the connectivity graph. Transitivity paths cannot be guaranteed by the *ad hoc* nature of the web-of-trust approach, so establishing a correct identity is not guaranteed as well. The next type of decentralised PKI is called the Statistical (quorum-based) approach. A public key is considered authentic when information can be obtained from many peers, where there exists a minimum called a quorum. The public key information is extracted from a statistical subset of multiple random and quasi-independent sources which confirm the public key. The hybrid approach uses a weighted quorum from a random set of independent peers.

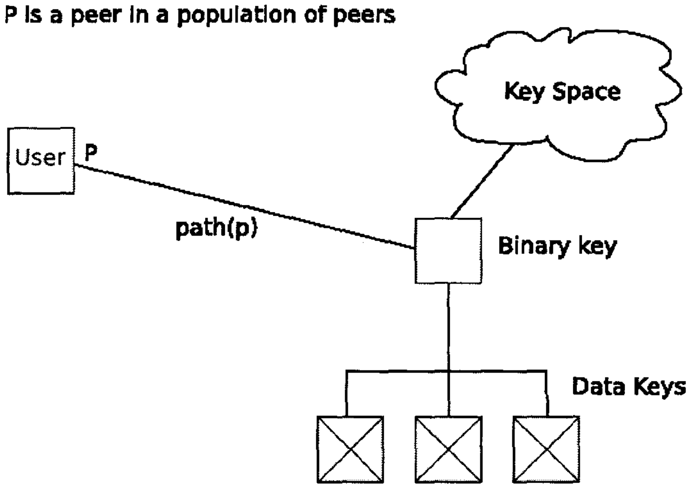

Figure 2.3: P-Grid based on [3]

Aberer *et al.* [3] advocate quorum-based decentralised PKIs by virtue of their usefulness in structured peer-to-peer systems. The authors present a system design based on P-Grid, illustrated in Figure 2.3, a decentralised PKI using Distributed Hash Tables (DHT). P-Grid associates peers with data keys from a key space, where a binary key determines the data keys a peer has to manage. A data key is associated with a binary key by way of a prefix of the peer. P-Grid is based on a distributed binary tree, where the search space is complete such that every key prefix is associated with a peer. By using a push/pull gossipping mechanism, co-ordination of stored and replicated data is probabilistically successful and guarantees consistency in unreliable network environments. Robustness is provided by each path being covered by multiple peers, replicating data and paths, ensuring retrievable data even if some nodes fail. A peer node is identified by an ID and an IP address with a time-stamp, where each peer has a routing table storing. The ID is generated by an algorithm when a peer joins the P-Grid community. A peer then stores a cache of known peers ID tuples along with the routing table. Updating of the cache and routing tables are achieved though work presented by Datta *et al.* [32]. This is done through a push/pull updating algorithm, in which a peer pushes an update to a subset of responsible peers it knows. This in turn propagates through the environment, since each peer will have a subset of other peers it knows about. When a peer joins the community, where the peer has been offline or disconnected, the peer enters the pull phase of the updating algorithm. In this phase the peer inquires for missed updates. The push request contains an update data item, its version number, a counter for the number of push rounds, and a partial list of peers this request has been sent to. A peer receiving a push request forwards the request to a random subset of the peers it knows. The version number and counter provide the history of the update and the latency of propagating the update, respectively. This way a peer will be able to find the most update data from the flood of replies.

SPKI

SPKI (also known as SPKI/SDSI) , in response to certain weak point in X.509-type certificates, incorporates a different naming scheme called a Simple Distributed Security Infrastructure (SDSI), presented in the RFC 2693 [36]. SDSI allowed the use of local names in a global name space, but since the merger of SPKI and SDSI, a public key is associated with a local name space [27]. RFC 2693 incorporates the S-Expression and naming scheme presented in SDSI with PKI, and represents experimental work [36]. A S-Expression is defined by a LISP-like parenthesised expression with a string identifying "type" followed by a value. SPKI name spaces are localised, where in a local name is a pair consisting of a public key and an arbitrary identifier. Public keys, called Principals, are used to sign statements, in this case certificates. A principal can be an individual, process or entity that are identified by the cryptographic key that represents them. There are three types of certificates represented by an S-expression:

ID - mapping <name, key>

Attribute - mapping <authorisation, name>

Authorisation - mapping <authorisation, key>

The focus of SPKI and the purpose of a SPKI certificate is authorisation, not authentication. Realising that SPKI will not become a standard within a short time period, the ability to translate from other certificate formats is included. This is particularly relevant when considering that the main purpose of SPKI certificates is authorisation. This concern is with *who* has valid access permissions to access *what* resources. Ellison *et al.* [36] give reasons why an authorisation should be limited to a single certificate by considering the X.509v3 extensions where an individual would be issued a master X.509 certificate with all the attributes and authorisations that are needed. If that were the case then the issuer of the master certificate would need to be the authority of all the attributes assigned to that certificate. Considering that each attribute has a certain lifespan, the result is the necessary shortening of the certificate's lifespan. Furthermore, privacy would be a concern where an individual's attributes would be lumped together, providing a possible breach in the individuals privacy.

An interesting feature of SPKI is the Access Control List (ACL). In RFC 2693 [36], the ACL is not standardised and is left to be implementation-specific. This is because the ACL is never communicated, and gives fine-grained control to the developers. The ACL is also given freedom to suite the needs of the system by directly designed to those needs. Furthermore, key management is handled by giving certificates a limited life-span, the end of which results in a new key pair being issued. This circumvents the need for an extensive key management system.

A feature of the SPKI approach is the ability to signify whether a certificate is transitivity, where certificate chains can be established and permissions delegated further down the chain. This is done using a propagation flag in the certificate, where a root "A" can signify that user "B" can re-delegate the permissions assigned by "A" to another user. It is important to note that SPKI certificate name spaces can be built "bottom-up", giving an alternative to the more traditional "root" hierarchy [27]. Two issues in terms of delegation are considered, the depth of the delegation and separating delegators from exercisers of delegated permissions [36]. Depth of delegation is signified by a Boolean value, indicating permission to delegation is true or not. A permission may be delegated to another entity or a subset of entities, with no control over the proliferation of delegations. This, however, also provides a weakness where a root of a chain has little control over subsequent user activities. A delegator may exercise any right it can delegate, in response to the second issue mentioned above. In addition to a delegation chain being a weakness, there exists issues with certificate-chain discovery, as addressed in [27]. In that, given a collection of certificates, it is the problem of establishing and constructing a suitable chain of certificates that will satisfy an ACL for a resource. The focus of Clarke *et al.* [27] is to solve it by presenting an algorithm that takes a set of certificates, the desired authorisation and the public key desired to prove that authorisation to solve the problem.

2.3 Access Control

Once identity is defined, and has a means of being represented and stored, mechanisms for controlling access can be developed. This is typically known as enforcing authentication and authorisation. The concept of trust can be shown to be dependant on the form and procedure of the authentication and authorisation mechanisms. Policies can be used to express and encapsulate the dynamic nature of an open environment, where decisions about authentication and authorisation are dependant on time and situational factors.

2.3.1 Authentication and Authorisation

IAM is there to control access to resources and services by a user based on an access and permission rules. In this view, authentication is the action of validating the identity of a user based on credentials and attributes that the user provides. Authorisation is the decision as whether to allow an authenticated user to access resources or services depending on the user's access rights. These are fundamental concepts to be explored [21]:

Authentication - positive identification of an entity seeking to gain access to secured information or services

Authorisation - an entity is granted a predetermined level of access to resources

Accounting - the use of each asset is then logged

Non-repudiation - using a trusted third party to verify the authenticity of a party's message

Mutual Authentication - each party during communication verifies the identity of the other

Multi-factor Authentication - using two or more factors of authentication

Access control decisions are encapsulated as a mechanism that seeks a resolution between requirements (expressed in policies) and the disclosure of proof (by an entity) that fulfil those requirements. A mechanism describes a process within a domain that has a purpose and will only operate within certain bounds [21]. A mechanism cannot deviate from its purpose unless it is explicitly controlled to that end or it is given the power itself to decide its own action. A mechanism that is controlled externally through policies receives its orders and context externally, and thus has greater flexibility. It is a matter of course to see that a mechanism that is externally controlled can adapt the process to suit the context.

An authentication and authorisation mechanism gains greater worth from being able to adapt to different contexts. External control in this case can be described through policies, where a policy is a set of rules and condition that can efficiently express a complex relationship between entities. Thus greater value will be gained by separating the authentication/authorisation process from the external control of policies. In that, authentication/authorisation decisions are based upon policies that are separate from the authentication/authorisation mechanism that enforces that decision. The value to be gained here is that the administrators and owners have fine-grained dynamic control of access to resources.

Authentication and Authorisation Infrastructures

Lopez *et al.* [77], introduce Authentication and Authorisation Infrastructures (AAI) and state that the challenge of an AAI is to provide an inter-domain authentication and authorisation service. The paper is an evaluation of existing AAIs in terms of certain criteria or features. These features are as follows:

Security - AAIs should be secure from the standpoint of identity and legitimate use of credentials, as well as protecting the stored data and data in transit.

Efficiency - in terms of computational and communication costs while attaining the goals of the AAI

Scalability - the AAI should scale to the intended environment, being a large, open distributed network

Interoperability - the AAI should be able to handle different formats of credentials and certificates, while not including proprietary information

Delegation - the ability of delegation should be handled in such a way to minimise user input, while managing delegation rights and permissions

Revocation - the AAI should have the power to revoke any rights or certificates issued

Privacy - the AAI should have control over the release of specific information, protecting its legitimate user's information

Mobility - authentication and authorisation information must become mobile, able to be incorporated by decentralised applications

Mobile computing - the previous requirement was for the mobility of information, this one is for the mobility of devices

In approaching a model for authentication, one has to decide where control decisions will be made, a centralised or a decentralised architecture. Ma and Woodhead [80] advocate the use of a decentralised approach called authentication delegation within a subscriber-based remote authentication scheme. In that, the resource provider delegates authentication to the subscriber based on digital certificates. Although this is similar to other centralised authentication infrastructures, it does not use a trusted third party and relies on the resource provider to perform the initial delegation. This is encapsulated in an authentication delegation certificate (ADC) where a *delagatee* (subscribing institution) public key is bound to a domain name (DN). The certificate is then signed by the resource provider to ensure integrity and authenticity. The relationship between the public key-DN pair provides a trust relationship, removing the need for a Certificate Revocation List (CRL). Access to resources are specified in authorisation policy certificates (APC), with resolution on the type of access, and other conditions and constraints. Ma and Woodhead make use of the authentication model specified in RFC 2753 [128], which provides a conceptual view of how to separate application independent policy decision points from application independent policy enforcement points. It is interesting to note that two modes of operation for the model exist in terms of the system gaining user credentials: one is where there is a credential push, in that the user gives credentials to the system. The other is a credential pull, where the system finds the credentials for the authentication process.

2.3.2 Trust

Trust is a critical aspect in any form of communication and co-operation. In the real world, we make a series of trust judgements during the course of our daily life whether we realise it or not. We put trust in other people to act rationally and trust in objects to perform their roles, and thus forms the basis for society.

A trust judgement is a subjective evaluation of the risks and consequences of the trust being broken (seen as betrayal) within a specific context where there exists uncertainty [46]. As illustrated in Figure 2.4, in the absence of uncertainty where as much information as possible is known, trust is complete and unconditional. Given for a certain level of knowledge, a more accurate trust evaluation and judgement can be made. Of course, there are things that may not be known that still affect the given situation and context, relevant to the issues of trust. Thus having blind or unconditional trust is not advisable. It is possible to view trust from a risk point of view, where a user or system assumes a level of risk when interacting with the environment [109]. Risk management is the process of identifying and assessing risk and employing mechanisms and procedures to mitigate that risk. Effective risk management requires removing uncertainty from a decision, where greater uncertainty might not necessarily mean greater risk but reduces the effectiveness of a risk assessment.

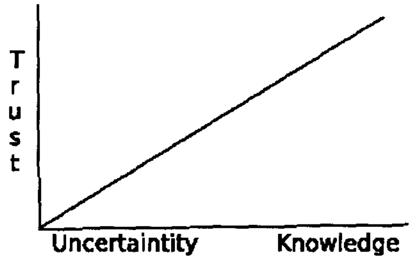

Figure 2.4: Trust versus uncertainty

Trust is especially important in the digital world, where sensitive data can be replicated or modified beyond the control of the owner. Since humans make a subjective value judgement, the concept of trust is not easily portable to the digital world. In the networked environment, trust is a concept that can have separate directions from which these concepts are defined. A metric for trust, an objective method for measuring trust, has to be established before trust can be expressed in any meaningful sense in a digital networked world. Firstly, the measurement of trust is dependant on the entities involved, for instance, within a community composed of human users it is difficult to evaluate the intentions and motives of the users where one can only make judgements about their behaviour and approximations of rational human behaviour. Secondly, one has to extend the concept of trust to machines since in a heterogeneous environment an entity may be human or machine, where there may not be enough free information to make a distinction.

Trust Metrics

Trust is a matter of subjectivity and becomes difficult to model. The knowledge domain of trust metrics tries to solve this by introducing an objective method for understanding trust in a digital world. However, there exists no general model for trust at the moment, where trust is specific to an environment and context. The modelling and measuring of trust becomes a matter of reducing uncertainty in an objective manner. With whatever strategy, trust will have to become systematic and measured as a metric.

There are two aspects of trust that need to be addressed: Firstly, trust management within a community of users. Secondly, trust management within a heterogeneous environment where it is difficult to establish the difference between user and machine. However, common to both aspects is the importance to establish a base level of trust of the machines in the environment. Consider the case of a user node in an *ad hoc* peer-to-peer network and the case of a user connecting to an on-line shopping site. Both users can transmit sensitive private information to another peer or the on-line shop site, where the transmitted information is out of the control of the user. Two levels of trust have to be monitored; trust of the users and trust of the machines. Trust in machines can be expressed as the compliance with the latest security programs and patches, and the latest defence-in-depth approaches and techniques. Securing individual machines, and the data contained within, can be considered to be an essential action in implementing defence in depth, regardless of where the machine's physical location may be. In a paper from the SANS Institute, Thomas Harbour [52] identifies areas essential to securing an end user's machine. The four layers of defence are identified and discussed; network access, Operating System, user applications, and the user's data. This translates to:

- Using an up-to-date personal firewall to control network level access to and from the machine
- Using a robust and hardened OS with regular patching, using the latest anti-virus software
- Patching applications with the latest updates, disabling scripting features of applications and email clients

- The backing up and encryption of critical and confidential data, using strong authentication methods

Employing measures as suggested above can, at a machine level, indicate enough information to generate sufficient trust levels in order to establish a base trust relationship to communicate sensitive digital data. Measuring a host's compliance and relating this to other hosts intending interaction can form a particular trust relationship. This can remove the uncertainty of dealing with a strange machine. As far as the author knows, there are no frameworks that incorporate defence-in-depth approaches in determining levels of trust.

A heterogeneous environment in terms of trust is where users interact with other users and machines (in this sense, any process that is not directly operated by a human), the user has to perform their own subject trust judgement before interacting. Fogg and Tseng [39] define terms and concepts relevant to users ascertaining the credibility of a computer. Here, credibility means believability in terms of perception over multiple dimensions of trustworthiness and expertise. Plainly put, a user is more likely to perceive that a website or an application is credible if it is dependable (trustworthy) and represents a wealth of knowledge and skill (expertise) in its knowledge domain. Unfortunately in the digital world, most users are clueless when it comes to the inner-workings of machines and applications and heavily rely on a visual presentation to judge their credibility. Empirical research quoted in [39] show that users who are ignorant have an inflated view of computers, provided computers act in the manner in which the user eagerly anticipates. The paper presents a model of how users perceive credibility from a system perspective and a psychological perspective. This, however, is limited to the users perception, in the way the user interacts with the system or application, and the system's operation and quality of its response. If an application has a security flaw that malicious people exploit, a user who is unaware may implicitly trust the application to performs its duties, while only after a length of time discover the negative consequences associated with the exploit.

User perception is one facet of trust in software, but trust should be considered from a developers perspective. Trusted components, re-usable software code that attains a specified level of quality and is assured of that quality [83], can aid in the assertion that a particular instance of software is trustworthy. In this sense, trust implies the belief that a program or a component of software (such as an API or library) will perform the documented actions. Furthermore, bugs, or undocumented and unintended features reduce the trust in a program or software. This type of trust can form the foundation for trust in machines, that a system or program will operate in the intended manner without compromising the user or the user's data. By following design methodologies that take secure and safe code into account, applying pre- and post-production analysis and testing through formal methods, it is possible to begin to assure the quality of components [83].

Trust within communities and between members of those communities are based on real world activities in society. This typically entails some form of accountability, where consequences are necessarily and invariably attached to actions of users. For instance, in a forum where registered users may post messages moderators may make executive decisions about the message should it violate the *"acceptable use policy"* of the forum. An extension of this is to rate the performance of entities based on their intentions or their expected behaviour contrasted by the results of the interaction. Perhaps a more illustrative example is of Ebay.com, which employs a reputation feedback mechanism that allows participants of a transaction, typically a buyer and seller, to rate each others performance [60]. Users can leverage the reputations and comments submitted by previous participants to aid in making transaction decisions. However, approaches to providing human generated trust measurement should be careful of being fallible to abuse. It should be avoided that a user who receives a negative rating may reciprocate the negative rating in an act of malicious retribution [50].

Advogato [75], a community website for open source developers, employs an attack-resistant trust metric. This approach maps each user account as a node in a graph where a certificate is represented as a direct edge. The community exists as a peer certification scheme, where a user is certified by fellow users, based on interactions. There exists three levels of certification, representing the different levels of participation, where a user gains a higher level through more interaction over time. The trust metric is run over three the levels, where the goal is to limit the number of "bad nodes" being certified by "good nodes". Using an algorithm to compute the network flow, from trusted node to trusted node, the shortest available path is taken. The attack-resistant property is given by separating nodes into good, confused (nodes that have certified bad nodes) and bad nodes, and showing that there is no flow from good to bad nodes. The remarkable aspect of this approach is its robustness in the face of a significantly massive attack, where the number of bad nodes that get through scale linearly when exponentially increasing the number of attacking nodes.

In an effort to remove the subjectivity of making a trust judgement, the decision to accept risk and proceed with interaction is encapsulated in the form and function of a system. This is explored in Trust Negotiation, discussed in Section 2.3.3.

2.3.3 Trust Negotiation and Management

Trust Management was first coined and expressed by Blaze *et al.* [16] in 1996 when recognising the need for a framework incorporating security policies, security credentials, and trust relationships. The work presented in the paper gave trust management semantic meaning separate from the previous systems or applications where trust was implicit in its operation. This paper represents pioneering work in distributed trust management, which takes a general framework approach focusing on the language of assertions rather than trust computations over the entire graph. The approach is based on several principles:

Unified mechanisms: common language for specifying policies, credentials and relationships.

Flexibility: scalability in large networks with the ability to succinctly and comprehensively express policy, credential and relationship information.

Locality of Control: each party/entity has control over whether to accept access of a second party (using credentials) based on their own policies.

Separation of mechanism from policy: the mechanism for verifying credentials does not depend on credentials themselves, using a single certificate verification infrastructure regardless of policy requirements and enforcement.

A brief overview of PGP and X.509 is presented in [16]. PGP has the notion that a security policy supports the verification of the ID of the sender of a message. These trust assertions are that the information is correct and not a statement of trust of the personal integrity of a user. This trust is not transitive, where each individual is responsible for forming their own opinion about the trustworthiness of the other users in the "*web of trust*". A note on X.509 is that it assumes and requires that the CA's form a global "*certifying authority tree*" and that all users within a "*community of interest*" have keys signed by CA's with a common ancestor. The Blaze *et al.* approach, PolicyMaker, is to bind public keys to predicates that describe the actions that they are trusted to sign for. This removes the PGP anarchic and *ad hoc* method of acquiring keys, and the X.509 centralised trust authorities. Trust relationships become more general and flexible (X.509 requires competing entities to enter into trust relationships). In PolicyMaker, security policies and credentials are defined in terms of predicates, called filters, which are associated with public keys. Security policies and credentials consist of a binding between a filter and one or more public keys. PolicyMaker was developed before web services became prevalent, meaning the implementation is more or less antiquated and has no usefulness other than forming a foundation for Trust Negotiation.

Trust Negotiation

Trust Negotiation (TN) extends the trust management approach by incorporating more criteria than just a unique identity [10]. This is portrayed through the use of formulated security policies and credentials that enhance meaning in a networked environment [85]. Basically, trust negotiation consists of a bilateral disclosure of digital credential, as in mutual authentication (thus avoiding man in the middle attacks and other phishing activities). By including mutual authentication, implying both entities have sensitive information they wish to protect, TN systems are closer to peer-to-peer structures than a client-server model. A trust negotiation protocol defines the sequence of message requests and replies, as illustrated in Figure 2.5.

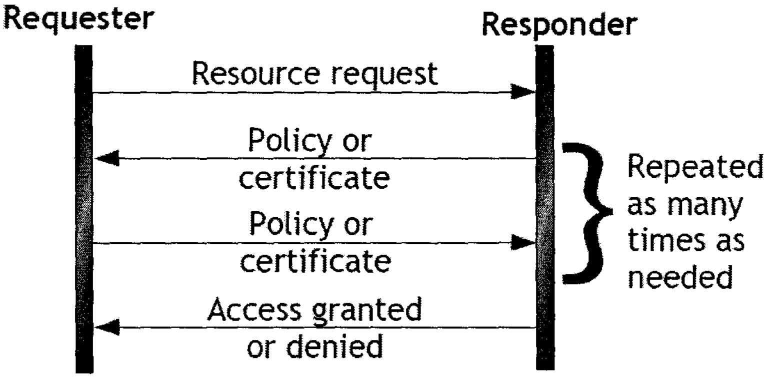

Figure 2.5: A typical Trust Negotiation protocol [85]

The basic components of trust negotiation system are entities (users, systems, processes, roles, and servers) and resources (sensitive information and services that have a set of policies protecting its disclosure) [85]. The typical interaction is a (client) entity making a request for a resource that is owned by a (server) entity. Credentials are stored in repositories, also called profiles. Additionally there are disclosure policies that define a set of access control policies. Policies themselves can contain sensitive information, where knowing the conditions and criteria of a policy can give the advantage to the attacker. Strategies are implemented as algorithms that define which credentials to disclose, when to disclose them and whether to accept or reject a request. Strategy efficiency is measured through its communication and computational costs. Digital credentials identifies and describes entities, and can be issued by owners of a domain or a trusted third party (CA). These credentials are considered to be sensitive information. Policy languages are a set of syntactic constructs and their associated semantics that encode security information exchanged during negotiations. The goal is to simplify credential specification that can express a range of protection requirements.

Winslett *et al* [123] in their paper introduce the TrustBuilder system for automating trust negotiation . The approach employs mutual disclosure of relevant credentials and policies between two parties. In the case of sensitive credentials and policies, a directed acyclic graph of policies can be used, meaning that an open chain of policies can be established where it doesn't cycle to the beginning. This protects the information and allows gradual establishment of trust. The concept of credential management is looked at briefly and examines issues surrounding credentials. Policies are thought to be best expressed in a policy language, where trust negotiation should be able to determine the satisfaction of a policy by a supplicant. For negotiation strategies to work in an open network, the same protocol must be used, a trust negotiation must either succeed or fail solidly, and dependencies for further disclosure must be track-able. The TrustBuilder architecture is explained, as well as its deployment.

Ryutov *et al.* [100]introduce a new adaptive trust negotiation and access control (ATNAC) framework. It explores the limitations to current ATNAC technologies, namely the GAA API and TrustBuilder. TrustBuilder is a middleware API that supports fine-grained access control and application level intrusion detection and response. The GAA-API allows dynamic adaptation to network threat conditions communicated by an Intrusion Detection System (IDS) [99]. The GAA-API can also detect some intrusions by evaluating access requests and determining whether the requests are allowed and if they represent a threat according to a policy. The GAA-API employs a policy evaluation mechanism extended with the ability to generate real time actions, such as checking a current system threat level, generating audit records and updating firewall rules. However, the API supports neither trust negotiation nor protection of sensitive policies.The focus of the paper is expanding upon current ATNAC schemes by incorporating features from both technologies that worked well, thus reducing the limitations faced by separate technologies. Their policies include those that govern public and sensitive resources, services, operations, and are expressed in the EACL (Extended Access Control List) format. For a request, the credentials are tested against the access control policies that govern access to the service. The credentials required are a function of the sensitivity of the request, the service or operation, system threat level and suspicion level. The suspicion level is a value attached to each monitored entity and is a measure of how likely the requestor is acting improperly. This is an implementation of trust negotiations and access control through governance policies.

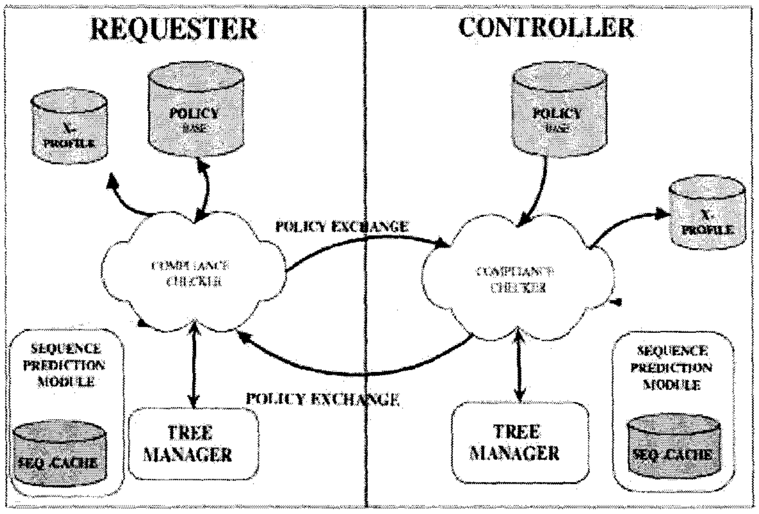

Figure 2.6: The Trust-X Architecture for Trust Negotiation [11]

Trust-X is an XML-based system addressing all phases of a negotiation and providing novel features with respect to existing approaches [11]. Its environment is peer-to-peer, in that all parties are equally responsible for negotiation management and can both drive the process by selecting the strategy that best suits their needs. X-TNL is a XML-based language used to describe and specify certificates and policies. A standard and expressive language for expressing security information is critical for the environment. It (the language) supports trust tickets, which represent a successfully completed negotiation that can be used in subsequent negotiations. Additionally, the language allows the specification and enforcement mechanisms for policy protection through policy preconditions. Each entity is characterised by a profile of certificates. A requestor is the entity trying to access resources, which negotiates with the controller entity. An owner has the resources while the controller may manage access. An entity may change roles from transaction to transaction. Mutual trust establishment is a feature of trust negotiation. Information release is governed through disclosure policies, which informs the other party the trust requirements required for the transaction. Figure 2.6 shows an overview of the main components of the Trust-X architecture for trust negotiation between peers. Due to the mutual trust negotiation sequence the architecture is symmetrical, where each peer has a Compliance Checker that resolves policy satisfaction and determines request replies. This component drives the exchange, drawing information used in the exchange from other components. The Policy Base stores disclosure policies. The X-Profile is the collection of certificates associated with the party (peer). The Tree Manager stores and tracks the state of the current negotiation. The Sequence Prediction Module stores information concerning recent successful trust negotiations, allowing the process to be sped up using previous negotiations. The disclsoure policies direct the trust negotiation sequence by defining the conditions under which a resource is allowed to be accessed.

As an intermediary step between current legacy systems, trust negotiation systems and the next-generation models, Traust gives trust negotiation technology space to grow and a chance for migration to trust negotiation [74]. Redesigning and re-standardising existing protocols takes time and effort as well as having negative effects on the widespread acceptance of new technologies. Traust acts as a third party authentication service that incorporates existing prototype TN systems (Trust-x or TrustBuilder: systems that allows clients to establish bilateral trust relationships with previously-unknown resource providers and negotiate for access in real-time) in large open environments. It integrates with newer, trust-aware resources while maintaining backwards compatibility with legacy resources. It can broker access tokens in any format, regardless of environment. It manages policy maintenance with overheads that scale independently of the number of users and their behaviour. Credentials and policies are seen as resources that are sensitive and have their own release policies. Typical trust negotiation implementations provide policy parsing, handling certified attributes, and determining policy satisfaction. Traust was designed with providing a general-purpose authorisation service which meet the needs of open systems to the highest degree possible in mind. Open systems, in the minds of the authors/designers, are seen to have the following requirements: bilateral trust establishment, run-time access policy discovery, privacy preservation, scalability (maintenance overhead and size of protection domain), and application support (relevant to Traust, not having to redesign etc). These requirements are intended for the specific needs of a system , and may be supplemented by others as the needs of the system changes.

2.3.4 Policy

Policies exist as a means to concisely and efficiently express rules and criteria that are used in decision making processes. RFC 2753 [128] defines a policy as *"The combination of rules and services where rules define the criteria for resource access and usage"*. The act of deciding to allow access of resources to a user can depend on a policy expressing certain requirements (such as the user's access level, or age). A policy can have varying levels of expressiveness; it may specify a high-level overview of how entities within a system may work or define their relationships. Policies also allow fine-grained low-level control over an entity's actions. When designing an IAM, it is important to take these different views of policies into account.

Criteria and Requirements

The following section introduces the requirement or criteria that are looked for in policy languages. It is important to note that in [10, 103] the language requirements discussed were in terms of a specific trust negotiation model. This resulted in some of the requirements presented as features required in a system implementation and could not be considered a feature requirement of the language. Each requirement is presented with a specific goal or reason in mind.

Well-defined semantics In [10, 103], policy languages are expected to have well-defined semantics that are simple, compact and mathematically defined. This means that the semantics should be able to concisely express policy and credential requirements in an unambiguous form and can be validated. The language should further be independent of the language's implementation and the platform on which it runs.

Representation of credentials and policies [103] suggests that the language must have means of constraining the values of credential attributes and the types of credentials. This suggests that the language should have a specification of the data structures for representing credentials and policies. Within the general model, a further requirement is that of the language being extensible with its expression of credential and attribute types.

Monotonicity A policy being Monotonic means that once trust has been established through the disclosure of credentials access should not be refuted through the disclosure of additional credentials [10, 16, 103]. This means that the language must be able to express and handle negative credentials, which fails the negotiation process if met. Further, monotonicity requires that trust negotiation is resolved quickly, or at least conditions where it fails are tested first. Although this is a feature of the system rather than a language feature, it illustrates the goal of monotonicity. It can dissuade certain Denial-of-Service attacks (E.g. wasting server time by engaging in needless rounds of trust negotiation).

Credential combination This requirement is that credentials stored with different repositories (or Certificating Authorities) and belonging to the same person are able to be combined to serve a particular request challenge [10, 103]. This allows users to be able to use different repositories to store confidential credentials. The user also has greater choice and is no longer limited to a single repository.

Inter-credential constraints As an extension of supporting credential combination, credential constraints and attributes must be able to span even though the credentials might have separate public keys [103]. Furthermore, this can be seen as the ability to merge policies or credentials (but can also be subject to policies) to yield a single policy or credential. This could add to the potential complexity of a system but it also allows greater flexibility of the system.

External functions The representation of strings and other basic data types, as well as function calls to standard libraries themselves should be platform independent and implementation non-specific [103]. This is a further requirement for the interoperability of the language.

Standard specification Interoperability increases the scope and usability of the policy language. This can increase the likelihood of the general model for trust negotiation and access control being adopted. The more prevalent a common system, the greater the flexibility and usefulness the system becomes. A standard represents a clearly defined scope or context in which to achieve communication between different hardware and software platforms [69]. Open standards are best suited in large, open and distributed environments where standards are ratified by technical committees. This process ensures that the creation of such standards meet the requirements and are fair to all parties involved. This is opposed to having an application-specific language where the language is specified by the application developers, and might not take all the parties interests into account. This also has the negative connotations of creating system/platform hooks where people are locked into a specific implementation or platform.

Policy Languages

The main focus of IAMs is the control of access to resources through advanced policies with the added requirement of detecting malicious or suspicious activity [98]. Policies are seen as a set of criteria or conditions that have to be fulfilled or met. Detecting incorrect behaviour characterises an adaptive policy. Other implementations are ineffective and not scalable, through policy reloading or changing the policy computation algorithms. These shortcoming are addressed in the paper of Ryutov and Neuman [98], by having a policy specification that describes more than one set of disjoint policies. Furthermore, the policy evaluation mechanism is extended to being able to read/write system state, allowing the monitoring and updating of internal system structures and their run-time behaviours. This requires, as a disadvantage, more tedious and careful policy specification, taking into account the side effects. The paper lists several conditions that make detecting anomalous activity or behaviour easier. This unfortunately requires careful considerations about the upper and lower bounds of these conditions, where some of the are rather contextual and trivial.

Ahn and Lam in [4] present a preference expression language used for specifying privacy preferences call PREP(PReference Expression for Privacy). The purpose of the paper is to illustrate the need for a standardised mechanism for stipulating a user's privacy preference. The proposed language, as presented through a "proof of concept" implementation, is used to store a user's privacy preferences with Liberty Alliance enabled attribute providers, with particular reference to privacy policies. Standardisation is an important aspect of this approach, especially when considering integration with an established standardised architecture such as Liberty Alliance. PREP is designed to be used by the attribute provider, and should be implemented by all entities in Circle of Trust. The attribute provider is required to capture a user's preference in order to store these preferences. This allows the attribute provider to effectively decide on attribute disclosure taking into account the level of the request and the policy to be used.

A different approach to constructing and managing policies is given by Lee *et al.* [73] by introducing a new concept called defeasible policy composition. Defeasible logic is modelled to mirror human common-sense reasoning, specifically a non-monotonic computationally-efficient logic. Non-monotonic means that a conclusion can be retracted in light of new information, allowing a revision of the conclusion. The authors define an abstract framework, where the focus is on policy composition and policy-related operations (such as combining several sub-policies). This approach allows policy writers to construct meta-policies, which describe the policy to enforce and annotations referring to policy composition preferences. It is important to note that meta-policies describe aspects about the policies, where those policies describe actual access controls and behaviour. The goals of this approach are to render policy writing in a human-readable manner, while allowing for the automation of combining different policies.

SAML 2.0

The Security Assertion Markup Language (SAML) is an OASIS specification and standard, which is a set of specifications for a XML-based framework for communicating user authentication, entitlement and attribute information. According to the website [41], this allows business entities to make assertions about the identity, attributes and entitlements of a subject to other entities. In order to strengthen business agreements and co-operation, identity is managed through federation. This is where a set of service providers agree on a way to refer to a single user , allowing Single Sign-On (SSO). The specifics of this type of identity management relies on the policies effected by the holders of the agreement.

SAML *"defines the syntax and processing semantics of the assertions made about a subject by a system entity"* [23]. An assertion is a *"piece of data produced by a SAML authority regarding either an act of authentication performed on a subject, attribute information about the subject, or authorisation data applying to the subject with respect to a specified resource"* [55]. There are three types of assertions statements that can be produced by a SAML authority entity:

Authentication - The assertion subject was authenticated by a particular means at a particular time.

Attribute - The assertion subject is associated with the supplied attributes.

Authorisation Decision - A request to allow the assertion subject to access the specified resource has been granted or denied.

Within the specification set exists definitions for the structure of SAML assertions and the processing rules for the management of SAML systems. Since SAML uses XML and XML names spaces to encode its assertions and protocol messages, SAML systems are not limited to the protocols used to communicate these assertions. The embedding and transport of SAML protocol messages are handled by SAML Bindings within a framework [23]. The framework incorporates a generic method for mapping a protocol message to some other protocol. Interoperability is achieved by specifying SAML profiles, which are a set of rules that govern the embedding and extracting of assertions into/from a protocol, the use of SAML protocol messages, and for mapping attributes expressed in SAML into another form of attribute representation system. Confidentiality and integrity are achieved by means of encryption, the use of which is also specified. A facet of SAML is that it embraces the idea of flexibility and extensibility of XML by allowing extensions to the assertion and protocol schema's. This allows implementations of SAML to incorporate domain-specific features and gives finer control to the developers.

XML-Based Policy Languages

XML-based specifications for policy, credential and authentication languages have flexible and interoperable virtues that make them suitable for an open environment. The use of XML in web services has been well documented and has contributed to the success of web services [7]. The use of the Dolev-Yao abstraction to model and formally analyse security protocols through their behaviour and properties of cryptographic operations in [7] further supports XML.

The Trust Policy Language (TPL) [53] developed at IBM Research is an XML-based language that is designed to define the mapping of strangers to roles based on credentials issued by third parties. TPL has well-defined semantics and portability due to XML. There are two types of TPL: Definite TPL (DTPL) and TPL itself. DTPL is a subset of TPL that excludes negative rules, and thus is monotonic. A group defines a role (for example, that of a clerk and a manager) in the system and represents a specific organisational unit. Membership to a group is controlled by a set of rules (for example, a manager can assign an employee the role of a clerk). The language can be mapped to a logic-type programming language for implementation.

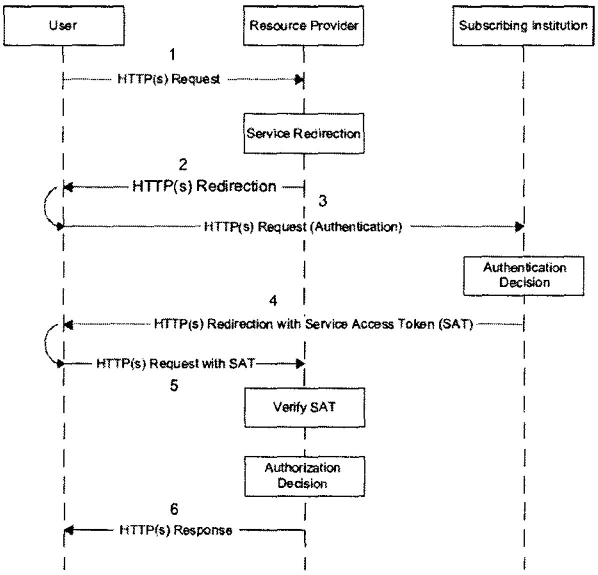

Figure 2.7: The Single Point Authentication protocol [80]

Ma and Woodhead [80] have an authorisation policy language based on XML, and a protocol based on SAML assertions to exchange authentication information. This follows the view that policy languages are used to conduct trust negotiations, via the gradual and incremental disclosure of credentials. XML represents a platform-independent mechanism to store data in a machine-readable format. It also allows the expression of precise definition of semantics, giving it flexibility with its extensible nature. With specific definitions, it is trivial to design a language for expressing policies and authentication information. This language is used in a Single Point Authentication (SPA) protocol based on SAML (See Section 2.3.4), which is similar to Liberty Alliance and Shibboleth protocols. Figure 2.7 depicts the SPA protocol between user, resource provider and the subscribing institution, noting that the protocol is run over HTTP(s). A user makes a request to the resource provider, which is redirected to the subscribing institution requiring the user to authenticate to the institution. If the authentication is successful, the user is issued with a Service Access Token (SAT) representing resource access permissions. The user presents the SAT to the resource provider, who in turn verifies the SAT with the subscribing institution and makes an authorisation decision. The SAT ticket is based on the SAML specification for making access assertions.

The eXtensible Access Control Markup Language (XACML) is a general purpose policy system, based on a strict subset of the SAML specification and is supported by OASIS. The purpose of XACML is to provide an interface for various systems and applications, integrating general policy access control functionality [79]. Policy systems are required to have policy processing and combination abilities, adaptable to a wide range of circumstances. XACML provides comprehensive policy management functionality in its specification, with all the advantages of SAML and XML. Policies consist of an arbitrary tree of sub-policies, where each tree represents a target and the leaves are the rules. A target in this case is a set of criteria used to determine a policy's applicability to a request. The rules contain complex logic making XACML extremely expressive and capable of handling a myriad of policy requirements. The policies affect the PDP and PEP of the authentication systems.

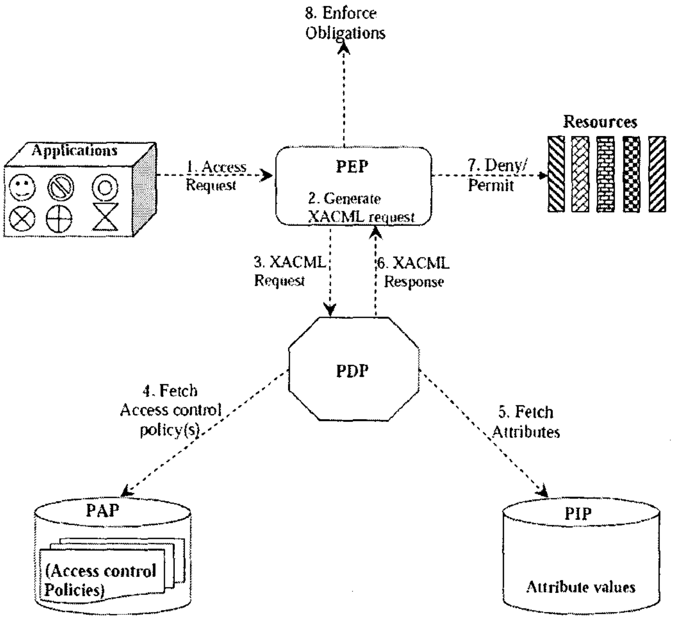

Figure 2.8: An overview of the XACML architecture [66]

Keleta *et al* [66], in Figure 2.8 give an overview of the XACML architecture, depicting the Policy Enforcement Point (PEP) receiving access request from applications. The PEP is responsible for enforcing decisions and obligations that are communicated from the Policy Decision Point (PDP). The PEP also enforces any obligations attached to the access policy at the time of decision enforcement. The PDP retrieves the access control policies from a repository called Policy Administration Point, which the only PDP is allowed to access. The PDP also retrieves attributes from an attribute repository, which stores user attributes and is basically another name for an Identity Provider. The PDP formulates a response by resolving the access request with the access policy and the user's attributes.

2.4 Chapter Summary

In this chapter, the concepts that comprise identity and access control were covered. This included different and often mutually exclusive approaches. However, from this chapter it is possible to see the components as separate and integral to each others operation and success.

Identity management is seen to depend on the name-space for which the system exists. The mechanisms for providing identity are further dependant on the environment in which the systems reside and the purpose and goals of the system. Within an open environment, federated identity is able to meet dynamic requirements, while still being developed and improved upon. There is no single over-arching solution to the myriad problems faced in an open environment, but rather, federated identity models are developed with specific goals in mind. Aspects of identity management that are taken into consideration ultimately depend on the needs of the organisation.

Access control is shown to be a divergent field with the overlapping of such concepts as trust, policies and protocol standards. Providing assured access to users and systems that have rights and privileges to do so is the main goal of access control. However, in achieving this goal, several different approaches can be employed. These are shown to have significant architectural and implementational differences.

The following chapter attempts to draw out and highlight issues that were briefly discussed in the problem statement (Section 1.2).

Chapter 3

Identified Issues

3.1 Chapter Overview

In the previous chapter IAMs were examined in two main areas, namely identity and access control. This chapter is an extension of the previous by further exploring literature under the three problem areas identified in the problem statement (see Section 1.2). These being the closely related privacy and anonymity, and pervasive multi-factor authentication.

Issues surrounding privacy are dealt with in Section 3.2. Privacy is concerned with protecting personal information and characteristics of users. Section 3.2.1 addresses privacy through the use of pseudonyms, a means of disguising identities using an alias. Identity information is disclosed during the authentication process. Section 3.2.2 discusses approaches to providing privacy during this process. Finally, Section 3.2.3 covers privacy laws and rights.

Section 3.3 refers to anonymity both in the identity of the user and within the communication channel. Anonymity is desirable in the face of passive traffic analysis, and subsequent directed attacks using the gleaned information. Section 3.3.1 explores some different approaches to providing anonymity.

Section 3.4 attempts to show the lack of a pervasive means to providing multi-factor authentication in an open environment for the typical user. The typical user, those with limited technical skills and who understand the minimum about a system to make use of it, is concerned about the user experience. Increased security measures impact on the usability of a system, detracting from the user experience. The section aims to promote the incorporation of the mobile phone as an ideal and pervasive security token.

Finally, Section 3.5 concludes and summarises the chapter.

3.2 Privacy

Digital privacy is the protection of the attributes and characteristics associated with a particular identity from being used outside the bounds of the subject's interests [122]. To a user privacy means the protection and the control of their own sensitive personal information.

Privacy is relevant in two areas, where the attributes and credentials are stored and when they are transmitted across an open channel. Privacy is achieved in these circumstances through the use of protocols, encryption and digitally-signed certificates [4, 13, 93]. Pseudonyms, the temporary use of a different name for the same entity, can add an extra level of privacy by decoupling a user's real identity and the identity used for transactions. Hence, there is a direct relation between privacy and anonymity.

3.2.1 Privacy Through Pseudonyms

Pseudonyms, as presented in [4, 29], require that the link between a pseudonym and the user's identity/account is not public knowledge and should remain hidden to be effective. There is a differentiation between transactional and situational pseudonyms. Transactional pseudonyms are short-lived and are generated for a single session or transaction. Situational pseudonyms are more permanent and serve as substitutes for user's real identity. It remains an aspect of any system that enables the use of pseudonyms as to whether attributes used in one may be transferable to another.

In Ahn and Lam [4] the main motivation for Federated Identity Management (FIM) is given as an enhancement of the user experience and privacy while decentralising user management tasks among business partners. The paper discusses user account management issues. Securing the communication channel and encrypting messages in given as a partial means to achieve privacy. A further measure is to obfuscate the messages by using the principle of pseudonymity. Since improper security measures may allow intermediaries within the communication channel to breach the privacy of the message. Pseudonyms are created so that they only have meaning in the relationship between the two communicating parties.

Djordjevic and Dimitrakos [34] make a distinction between the way in which identities are represented. The first is the regular user-name mapped to a user's identity. The second is a pseudonym-

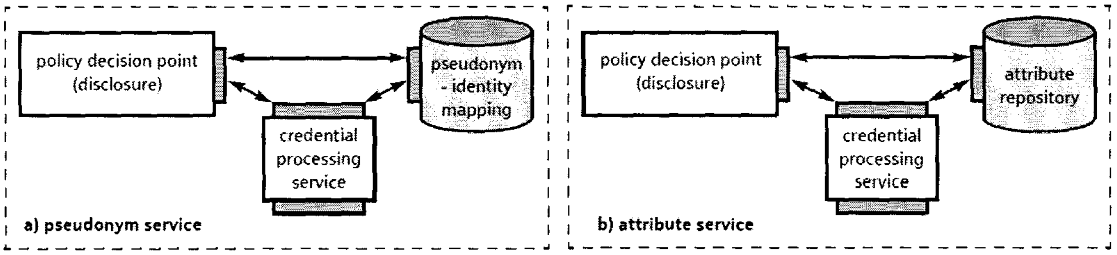
policy decision point (disclosure)
pseudonym - identity mapping
credential processing service
a) pseudonym service
policy decision point (disclosure)
attribute repository
credential processing service
b) attribute service

Figure 3.1: A Pseudonym and Attribute Service [34]

based approach, where identification information is obfuscated from Service Providers, avoiding unnecessary correlation between instances of that identity. The third approach is that of attribute credentials, where an identity of an individual is made up of a collection of attributes. This is particularly useful when Service Providers and Identity Providers establish a back-channel with security token services to communicate the authorisation message request-response sequence to facilitate single sign-on. Refer to Figure 3.1, which illustrates pseudonym and attribute services interacting with PDP and CPS components.

3.2.2 Credential Disclosure

Credential disclosure is an integral part of Trust Negotiation. It is the process of releasing required identity-specific credentials in order to achieve the desired authentication or authorisation. This process is typically encapsulated in a protocol, as discussed in Section 2.3.3. Privacy is an aspect that is usually out of the direct control of the user. Once private and confidential user information is transmitted across an open channel the user has no control over what happens to that information. Therefore special care has to be taken with the release or disclosure of user information.

In Bhargav-Spantzel *et al.* [13], trust negotiation's ultimate goal is to handle introductions between strangers in an open environment. This is achieved though the bilateral disclosure of credentials, where credentials are assertions stating one or more properties about the entity. Credentials are certified by the issuing authority of the entity's identity. These credentials are considered to be confidential themselves. Trust negotiation can be used to establish the vendor's credential requirement in order for a customer to access a resource. These requirements can be expressed in policies, which is a set of rules specifying conditions and criteria constraints of credential attributes. Additionally, policies can effect the disclosure process, specifying the minimum required user attributes to satisfy a request. This negotiation process is usually a set of challenge/response messages defined by a trust negotiation protocol that gradually disclose further credentials until the request is satisfied.

Cameron Morris in [85] defines the problem statement of his Masters Thesis as: "*Using certificates as digital credentials for users limit the adoption of trust negotiation since 1) users rarely seek out and obtain certificates, 2) issuers rarely issue certificates suitable for trust negotiation, and 3) obtaining certificates is inconvenient for users.*". The approach Morris took to solve this problem was using a Browser-Based Trust Negotiation (BBTN) scheme, where a central credential authority negotiates on behalf of a user as a proxy for web-based transactions. This approach ensures negotiations can occur from any machine connected to the Internet without the need for specialised client software. Additionally, the user experience is made easier by the central authority storing user credential information. The user is required to log-on to the central authority site to make use of its services.

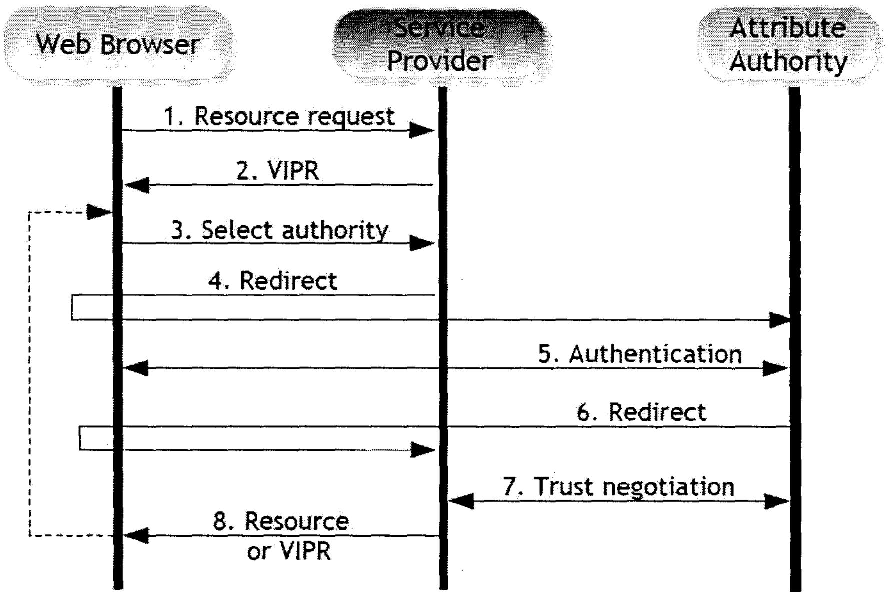

Figure 3.2: The Browser-Based Trust Negotiation Message Flow [85]

Figure 3.2 demonstrates the typical message flow used in the BBTN scheme. The user makes a request for access to a resource from the browser to the Service Provider (SP). The purpose of VIPR (Visual Policy Resolver) service is to present the user with choices of Attribute Authorities (AA) and the required attributes in order to satisfy a request. The VIPR presents a visual representation of the policy tree generated for a request, displaying the progress of the negotiation. Should a request fail, the VIPR is used to indicate the reason in terms of what attribute is required from a specific authority. The SP redirects the user to the chosen AA, to which the user is required to to authenticate. The SP and AA mutually authenticate each other in the Trust Negotiation stage using a back channel, as is the norm for SAML-based implementations. Figure 3.3 illustrates the SAML TN protocol used in steps 5 and 7 of the previously mentioned figure (Figure 3.2). After which, the user is presented with resource in success, or another VIPR page following failure. In a web environment, the ideal technology for defining a protocol and server-side system is SAML (Security Assertion Mark-up Language), as covered in Section 2.3.4. The BBTN scheme extends the SAML protocols by including a *NegotiationMessage* request, used in Figure , which defines the request and the response type in terms of either assertions or policies.

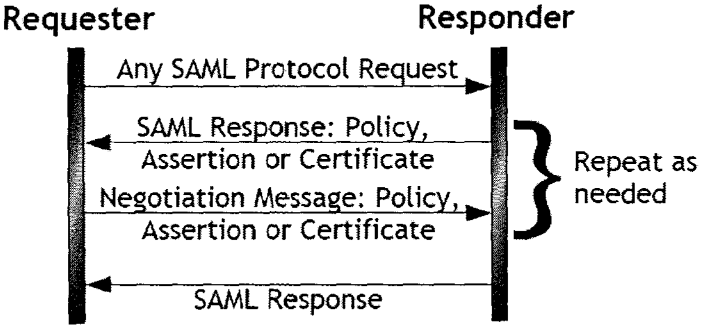
Requester
Responder
Any SAML Protocol Request
SAML Response: Policy, Assertion or Certificate
Negotiation Message: Policy, Assertion or Certificate
Repeat as needed
SAML Response

Figure 3.3: The SAML Trust Negotiation Protocol [85]

Another method in trust negotiation is to eliminate uncertainty by employing controls at critical points in a system and addressing identity issues, as found in Shibboleth [59]. Shibboleth is an initiative to develop an open, standards-based solution to meet the needs for organisations to exchange information about their users in a secure, and privacy-preserving manner. It is focused on local authentication and authorisation control for remote web-based connections to specific resources. It makes use of several widely-implemented standards such as SAML, SSL and LDAP [41]. Using an open standard like SAML, it is possible to monitor and provide information about the general flow of information within protocols. Further, the use of SAML is limited to the transmission of information. Shibboleth provides its own infrastructure and trust framework to work in conjunction with SAML. The Shibboleth Architecture has several functional components. Firstly, there is an Identity Provider (IdP) that maintains user credentials and attributes [41]. This is further made up of Authentication Authority, which is responsible for issuing authentication statements to other components. The Single Sign-on (SSO) service initiates the authentication process and is the first point of contact for the user, and interacts with the authentication authority behind the scenes to provide the necessary authentication assertion. Since Shibboleth is primarily web-based, it makes use of SAML Artifacts. These are small, fixed-size, structured data objects pointing to a variably-sized SAML protocol message and is embedding in URLS and conveyed in HTTP messages [55]. The Artifact Resolution Service is used to retransmit artifacts sent by the Service Provider (SP) via a back-channel exchange. This occurs because the IdP usually sends an artifact to the SP instead of the actual assertion. The Attribute Authority issues attribute assertions in response to attribute requests. Each request is authenticated and authorised by the attribute authority. A SP manages secured resources and allows user access based on assertions received from the IdP. The Assertion Consumer Service processes the authentication assertions returned by the SSO service or artifact resolution service, initiates an optional attribute request, establishes a security context at the SP, and redirects the user to the desired resource. This is the SP endpoint in the SSO exchange. A security context is defined by security models and a system architecture, where a set of system entities are authorised to access a set of resources [55]. This can also be seen as a semantic union of all the security mechanisms employed across the network. An Attribute Requester of the SP and the attribute authority of the IdP conduct a back-channel attribute exchange once a security context has been established. Policies can be expressed as a set of rules for these attributes. The reference model uses XACML policies to decide on access. Access control is split into policy decision and policy enforcement, using attributes and policies to decide on access rights. It depends on the structure of the AAI, its purpose and higher level policies as to how the AAI will deal with a user's personal information, both in storage and processing for determining access rights. Privacy then depends on these decisions, rather than becoming a feature of the infrastructure.

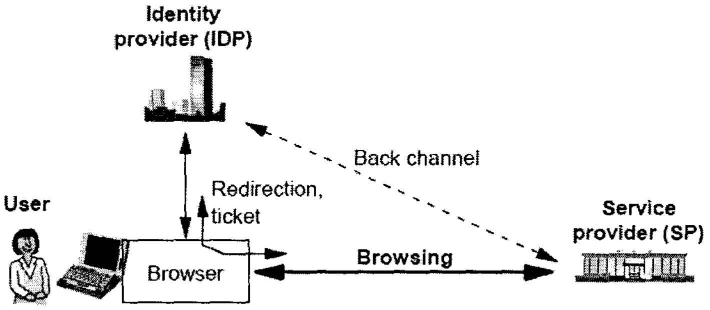

Figure 3.4: An example of Browser-Based attribute exchange [92]

In [93], the basis for the paper is privacy in browser-based attribute exchange, where a user can have multiple wallets associated with a single real-world identity. A wallet is defined as a collection of attributes that identify a user's identity or role. As a set of privacy requirements, users are seen to have direct control over storage and usage of their personal information. By having a wallet holder store their wallet, effectively storing their personal information, it should be done in such a way that it assures the user of privacy. Even though attributes are stored in relation to a user's unique identity, this alone does not assure privacy. By allowing the user to create multiple wallets and roles for different purposes gives more control to the user. Additionally, by allowing the user to choose their wallet holder (typically seen as a IdP) empowers the user to make informed decisions about where to stores their information. This, however, assumes that the different IdP publish their privacy policies and practices. To further assure users, as well as to make this architecture viable, constraints are placed on the wallet holders. Firstly, traffic to and from wallet holders should be done by consent of the user. Secondly, mechanisms are placed such that the wallet holders glean no additional information about the user from its operation. A back-channel can be established to ease the amount of input the user has to provide during authentication and authorisation. Illustrated in Figure 3.4 is an example of how the Browser can interact with a SP and IDP, easing the user experience.

In Pfitzmann and Waidner [93], privacy is seen as a major requirement for users and is a big concern for browser-based on-line business and personal transactions. In large open networks such as the Internet, mobility and remote access work well with the browser approach. User attributes contain personal information which has to be stored somewhere. Attributes can be required by destination sites to process certain transactions or access certain resources. The paper incorporates zero-footprint, a concept within browser-based attribute exchange, where the user's information is not stored locally. This is a feature of the environment, for instance, when a user attempts to access services from a public Internet kiosk. A requirement for the use of zero-footprint is that it should not be a hindrance to the user experience and should work seamlessly with the protocols and the back-end server.

3.2.3 Privacy Laws and Rights

As a South African citizen, the constitution provides for an individuals right to privacy [58]. This entails that a person should have control over their personal information, with relative freedom to pursue their own affairs. This relativity of "relative freedom" regards being within the confines of law. The law itself is not fully developed when concerning digital privacy, and the privacy of digital data. As such, until defining legislation is passed concerning information privacy, privacy rights encompasses both the digital and real world. This is recognised as a fundamental right of being an individual within the constitution.

Sites publishing privacy policies so that users are aware of the privacy practices enables the user to effectively decide whether or no to use the advertised services. In [4] four key principles of fair information practices are summarised. These are notice, choice, access and security. Notice requires that the vendor clearly display and notify users of information practices before the user engages in any activities. The user should be given a choice about which information to disclose and how it is used. Users should be able to access and modify their own personal information. Finally, the users should be assured of the protection and securing of their personal information that is stored there.

For eCommerce, preserving privacy and customer's data security is contrary to the vendor's needs. As presented in [101], through the author's research, a trade-off between customer and vendor privacy needs have to be made. The vendor requires to have access to certain customer information in order to conduct on-line business. The customer requires privacy in order to mitigate the detrimental effects of confidential information being leaked or disclosed. Although the customer and vendor can enter into a transaction, the customer must effectively trust the vendor not to release information disclosed during the transaction. The same remains true for the vendor having to trust the customer. For instance, if the vendor had to offer a discount to a specific customer and not wanting the other customers to know since they might ask for a discount as well. This matter of bilateral trust then becomes circumstantial, depending on the nature and details of the customer/vendor relationship.

Ownership and Control

The ownership of accounts depend on the organisation's "terms and agreements" or acceptable use policy (AUP). In most cases ownership resides with the organisation that supports the account, where a user is given an account if they are affiliated with an organisation. Use of that account is subject to the terms and regulations of the organisation. In cases where the user owns the account there is generally some form of monetary exchange for services, while still being subject to service agreement. In a general model, control over accounts can be expressed and enforced through the use of policies. In the case of registration in exchange for free services, the organisation creates accounts on behalf of a user, yet retains ownership over that account, giving superficial control to the user. This point relies on the fact that whoever stores the account information may well own the account itself and provides the use of the account and related services as a service in itself. If the organisation closes down or is bought out by other organisations, it is not clear what will happen to the information contained in that organisation. Consider Microsoft, if they were to be bought out by some global concern, the new owners could do whatever they like with the masses of customer information they have captured over the years of Microsoft's operation. Due to the "Privacy" and the "Terms of Use" policies that Microsoft [30] employs, in which it is stipulated that, they [Microsoft] have the power to change it at any time without customer consent. This supports the thought that Microsoft ultimately owns any account created though the use of their products. In fact, as users, people have to trust the organisation that they engage with will continue to operate in the user's best interests. This requirement of trust can be mitigated somewhat by the organisation publishing a legally-binding contract that specifies the clear constraints of operation of the accounts and user information. By having the organisation held accountable for any misuse of a user's private information, users are more likely to trust that organisation.

One such approach hands matters of identity to Microsoft to control. In Bahl *et al.* [8], privacy is attained by using a globally available database for credentials that users can authenticate against. During authentication a dynamically generated varying-length key is used that is valid for specific length of time. Although using a global authenticator, in this case it is MS-Passport, there is a Protocol for Authorisation and Negotiation of Services (PANS) Authorizor which generates the key/token pair once the user is authenticated against the global authenticator. Since this key/token pair only exists for certain length of time, if an attacker were monitoring the communication stream, the potential for attack would only last for that session. This also has the effect of decoupling the user's identity with the session identity.

3.3 Anonymity

Anonymity is seen as a means to protect the identity of the participants in a certain event from being known [15]. This also has the benefit of bestowing privacy on the participants. However, a distinction must be made between privacy and anonymity. Privacy, achieved through making the contents of a message confidential, is different from anonymity, where the focus is to hide the identity of the sender and recipient.

Kesdogan and Palmer define three aspects of anonymity, where anonymity is a direct trade off between accountability [67]. Being accountable entails the ability of linking actions with the participants of that action.

Anonymity - the state of being not identifiable within a set of subjects, where within communication it is extended to receiver and sender anonymity.

Unobservability - the state of an item being indistinguishable from any other item.

Unlinkable - of items and actions are no more or less related than they were before.

Some weak points in anonymity are discussed in [67]. It depends on others to work, in that within open environments there is no or little control of the channel for the participants in communication. Furthermore, Pfitzmann [90] relates that there are several possible attackers, including and not limited to "the administration, foreign states, companies, one's neighbours and communication partners". A second weak point arises from long periods of watching the behaviour of users where patterns can be discerned and knowledge can be inferred. If an attacker could observe a message exchange from a sender to a receiver, anonymity and unobservability cannot be achieved. The attacker is able to infer the communication relationship between the sender and receiver.

Clauss and Kohntopp [29] note that the eCommerce domain lacks anonymity and authenticity. A user leaves traces while using the Internet and that communication data can be faked or spoofed. Here anonymity is a trade-off between authenticity, which is required to achieve a trust-enabled environment. However, there is a similarity between the terms authenticity and accountability. In both the system attempts to determine the true origination of a message. However, it is relevant to note that true anonymity is not desirable by the fact that attackers may be able to commit fraud, so there has to be a balance between the ability to enforce accountability and protecting the identity of the individual.

3.3.1 Approaches to Providing Anonymity

In this section, we cover some approaches to providing anonymity. Traffic analysis is used to discern services within a network [94], and to reveal communication relationships. The anonymity of communication connection and message is resistant to traffic analysis.

Anonymous routing covers means to obfuscate the communication link between parties. Blind signature is used to assure the authenticity of a message without compromising the message to the signing party. Zero-knowledge proofs provides an anonymous mechanism to prove the knowledge of the existence of a secret without revealing the secret.

Anonymous Routing

An approach to anonymity over the communication channel was developed in the 1980's [90] called a MIX. However, this approach was limited to a closed environment where the number of users is known and rather small (around a 1000). Kedogan *et al.* [65] have worked to move this concept to be able to work within an open environment. This technique assumes that there are observers on the channel that have access to every single packet transmitted, and that the attacker cannot break the cryptography employed by the system. A MIX is a system that accumulates packets from a set of distinct users that seek anonymity of their communication. This set is known as the anonymity set because the participant that can link an input packet to an output packet is limited to the MIX and the sender of the packet. The accumulated packets are called a batch, where the bit pattern of the packet and the order of incoming packets are altered. Each packet sent to the MIX by individual users are encrypted specifically for that MIX by the user, where it is decrypted and altered. The MIX is aware of individual users and must link the input packets to a specific user. As such there should be an anonymous loop back so that users can verify their packets. A series of MIXes are used to increase the security and anonymity of the system, and so the anonymous loop back mechanism has importance, since the first MIX employed is only one that link sender and packet. Once a batch is accumulated and altered, they are dispatched to their respective destinations. Any responses come back through the MIX and are forwarded in the same way back to the original sender. Here, the MIX performs the same role as a proxy, providing a "black box" between sender and receiver, incorporating different variables into the communication channel. Since an observer, over a long enough time period, can infer information about the participants using a communication channel, it is necessary to incorporate a sense of randomness. Less can be inferred when what is observable is less dependant and determinant on other observable factors.

Onion routing, Reed *et al.* [94], is an approach that attempts to protect Internet services against eavesdropping and traffic analysis. In such an environment as the Internet, services require protection from within the domain network and from outside. The approach entails removing the direct socket connections between machines by providing a series of machines, called onion routers, for communication. In order to send a message to a responder, the initiator accesses the onion network through a series of onion proxies. An onion proxy receives connections, whose purpose it is to define a routing path through the network. This routing path is constructed into an onion, a layered data structure, where each layer is encrypted, containing the next destination. Each communication hop within the routing path is encrypted with different keys so that the onion appears different to each router. When the onion is passed to the entry funnel, the first onion router in the sequence peels off the top layer by decrypting it, yielding the next destination with the encrypted payload. The onion is routed until the last layer has been peeled off, yielding the true destination and payload, which is then passed to an exit funnel. Subsequently, an anonymous connection for a data stream is created using that routing path.

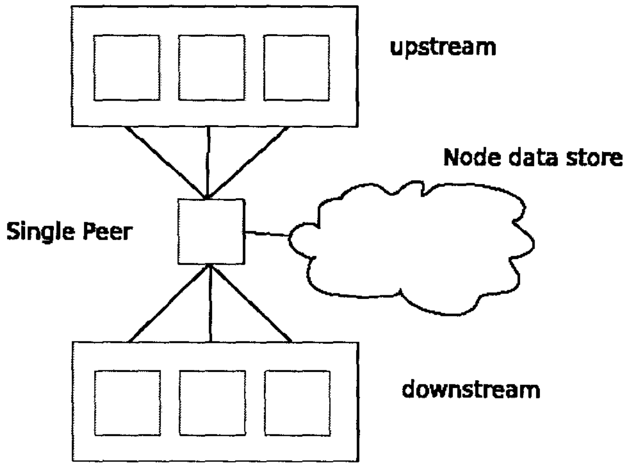

Figure 3.5: Freenet based on [28]

Clark *et al.* [28] describe another peer-to-peer network application called Freenet that anonymously stores distributed data for publication, replication, and retrieval. This approach protects the privacy and anonymity of both the data and users. It is an adaptive system that responds to usage patterns, such that files are moved, replicated and deleted as necessary. The approach is depicted in the simple diagram of Figure 3.5. It incorporates a novel idea by having users share their unused hard-drive disk space such that the user has no control over what is stored. The data itself is encrypted which is made available to the network for reading and writing. Each node has its local data-store and a dynamic routing table, which contains addresses and keys of nodes. Queries are passed from node to node in a chain of proxy requests. A node in a chain only knows its immediate upstream and downstream neighbours, and decides where to send a request upon its receipt. A query has a time-to-live count as to prevent infinite chains, it also has a pseudo-random identifier so that nodes can discard queries it has seen before. Results of the query are passed back, and is given in the text as a steepest-ascent hill-climbing search with backtracking. This means that the query is propagated to a node until either a result or failure is found. Security is a major aspect of the Freenet with goals of protecting the anonymity of requestors and inserters of files, and the identity of storers of files. Files themselves must be protected against malicious modification. Observers of the channel, those that manage to decrypt each packet, will not be able to infer the source of a particular data-store. This is achieved through the occasional intermediate node resetting the data source field during a reply. Perhaps through long-term observation of successive failure or success replies to queries, it is possible to infer a source of a data-store. Though this can be mitigated by the reshuffling the distribution of data files over the set of nodes.

An interesting taxonomy on anonymity is presented by Reiter and Rubin [95], which contains the design and implementation of a system called Crowd. The context for this system is on-line web transactions. An attacker can know the IP addresses of the client and server machines, and other information about the transaction, even though technologies like SSL to protect the communication data are used. The notion of degrees of anonymity is explored, where on one end of the spectrum is absolute privacy and the other is *provably exposed*, with four degrees in between. Absolute privacy assures that an attacker can by no means observe any evidence of communication. *Provably exposed* is when an attacker can provably identify both parties in communication. The other degrees in between range from strongest to weakest: beyond suspicion, probable innocence, possible innocence, and exposed. The approach is to conceal senders and receivers in a crowd of other users that are indistinguishable from each other. A user is seen as a node amongst a crowd of nodes. Web servers are there to respond to requests from the nodes. The user starts the Crowd application called jondo, which gains the user access to the Crowd network. The jondo picks up browser requests (during the course of normal web activity) from the user and randomly picks an initial jondo from the crowd to forward the request to. Conceptually, the initial jondo then picks a random path of jondos of a random length to and from the web server. The mechanism is that the initial jondo decides (as in a coin toss) whether to forward the request to another random jondo or to a web server. This mechanism occurs at every jondo that receives a request, where each jondo acts as a proxy. This approach does have increased message overheads that increase as the path length and packet size increase.

Anonymity can be provided by having a proxy that acts on the user's behalf that utilises anonymous attribute certificates, as indicated by Schlager *et al.* [101]. A temporary pseudonym can carry authorisation information specific to that session once the user has been authentication against the Identity Provider. However, true anonymity may not be achieved when certain information that can identify the customer is required to complete the transaction between a customer and vendor. For example when the vendor requires a shipping address, a customer's identity can be inferred from such knowledge. But to anyone observing the communication channel there should be no discernible user information contained in the messages.

Blind Signature Protocols

Effectively, through the use of encryption and signature protocols it is possible to attain a level of anonymity [25]. However, the level of anonymity depends on the strength of both the cryptography and protocol, and the parties involved in the communication. Blind signatures enable a message from a sender to be signed by a signer without revealing the message itself or the signature [107].

Blind signatures can allow fraud to occur because the anonymity provided can screen attacker's actions. But since it is susceptible to fraud, the authors of [107] further extend the approach by incorporating a third entity (a trusted third party that acts as a judge) and a second protocol. The first protocol is a signing protocol and is between the signer and the sender. The second protocol is link-recovery protocol that is between signer and the judge. The signer obtains information from the judge, enabling the signer to recognise the corresponding protocol view and the message-signature pair. From this, two types schemes are derived:

Type I: Given the signers view of the protocol, the judge delivers information that enables the signer or everybody to efficiently recognise the corresponding message-signature pair

Type II: Given the message-signature pair, the judge delivers information that enables the signer to efficiently identify the sender of that message or to find the corresponding view of the signing protocol

Figure 3.6 presents a standard model for fair blind signature schemes, involving the components presented above. During the course of the paper, several encryption-based fair blind signature schemes that conform to either Type I or Type II, or even both, are covered.

In [9], the authors use a scheme presented in [107] that incorporates both Type I and II though the registration of two pseudonyms with the judge. There are four parts to the protocol used to obtain and use an attribute certificate in [9]. Firstly a user requests a pseudonym from the Trusted Third Party (TTP), where the pseudonym consists of a public and a separate private part. The TTP stores both parts as a linked pair, in case the relationship needs to be disclosed. Both parts are signed and assigned a purpose and validity period and then sent to the user. Secondly, the user creates a new asymmetric key pair to be associated with the attribute certificate. The public key is fair blind-signed by the authority, based on the public pseudonym, the set of proofs (items used to fulfil the requirements, usually attributes signed by an authority) and information about the TTP. The authority decides (usually based on policies) whether to issue a certificate

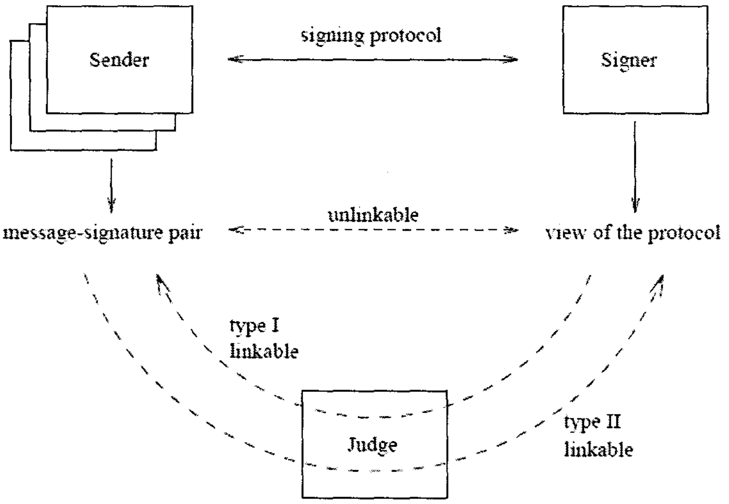

Figure 3.6: A model of a fair blind signature scheme [107]

containing the user-request attribute. The authority stores the attribute before sending it back to the user as a signed message. The user then transforms the signature into one using the public key over the private pseudonym. Thirdly, The user sends the fair blind signature, proof that it is valid, and the structure created based on part two to the AA. The AA issues an attribute certificate with a set period of validity and sends it back to the user. The TTP can reveal the relationship at this point when the specified condition (contained in the structure) is met. Fourthly, in order to use the attribute certificates, the user sends it and the public pseudonym to a SP. The SP is able to challenge the user for a signature to prove ownership. Any misuse or problems found, the SP can send the information to the TTP and AA in order for the pseudonyms and anonymous attribute certificates to be revoked. A snapshot of a system overview is provided in Figure 3.7, where the Source of Authority (SOA) forms the root of delegation chains for the Attribute Authority.

In [9], an approach to providing anonymity is introduced by combining X.509 attribute certificates and fair blind signature schemes. The authors extend the certificates by defining Anonymous Attribute Certificates in which the user's identity can be traced only under certain conditions. A certain field in the X.509 certificate, namely Holder, is now represented as a hashed function of several fields pertaining to the pseudonym used. These pseudonym fields are continued in another structure called the pseudonym structure and is hashed to form a digest. The hash value is then stored in Holder field.

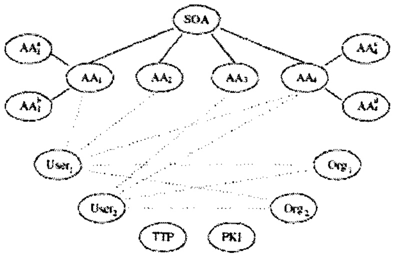

Figure 3.7: A fair blind signature system overview [9]

Zero-Knowledge Proofs

Zero-Knowledge proofs are defined in Goldreich [47] as a convincing proof that yields nothing beyond the validity of the assertion being proved. In that, given two parties where Alice, knowing a secret M, attempts to prove to Bob that she does indeed know the secret without revealing the secret to Bob. This approach becomes feasible cryptographically with the use of one-way functions and a predetermined protocol of behaviour for the two parties.

Zero-knowledge proofs can be used in anonymous authentication schemes, as presented in [116]. Though one approach can only work with a common trusted third party between the user and the service site. This is similar to the scheme presented in [9]. Where a TTP registers protected attributes of user which is used in the anonymous authentication process. The paper looks at the cryptographic techniques behind public-key certificates and witness-indistinguishable proof systems (as in zero-knowledge proof). Proof of correctness is provided through theorems. Time dependent hierarchical key assignment scheme is based on the hardness of computing the *e*th root modulo a composite and the common-modulus property of modular exponentiations.

In [14] there is emphasis placed on the creation of unique identifiers for each user and classifying the attributes of that user. There are uncertified attributes, information that is given freely by the user. Then there are certified attributes, which have been verified and issued as signed digital certificates by trusted SPs or CAs. Finally, there are attributes "secured against identity theft" (SIT attributes), which are used for user identification and are secured by the protection methods. Bhargav-Spantzel *et al.* [14] look at issuing an identity account based on the user's real world identity, defining a root identity before certifying the user's attributes. The user can decide which SP they will use as a registrar by storing their SIT attributes with that SP. This is inherently a distributed architecture that is consistent with the distributed protocols deployed in federation environments. The registration of SIT attributes (the first step in attaining a federated identity) is through a bootstrapping process. SIT attributes have the quality of their use being restricted unless there is proof of additional identity provided. This process is defined as a individual going to a SP and presenting physical proof of identity from which the SIT attributes are derived. The officer at the SP signs using his identity and asserts that the individuals information is correct. Further attributes can be stored via on-line methods using zero-knowledge proofs to prove the user's identity without actually revealing anything. To protect against using unauthorised unique identifiers (such as credit card numbers or social security numbers) a distributed hash table mechanism is proposed , using the SPs to storing and retrieving these values. However, it is important to note that the authors see the federated system as a closed system with inherent trust between the different SPs. It is also assumed that the SPs themselves will not be compromised in any way.

3.4 Multi-Factor Authentication

In Section 2.3.1 we covered concepts of authentication and authorisation. In this section we delve into the short-comings surrounding factors of authentication. Multi-factor authentication is devised as two or more of the following: Something you know (as in a password), something you have (a token), and something you are (biometric information). However, tokens and biometrics severely impact on the freedom of the user and the usability of the system. Biometrics are considered physically intrusive and are currently not suited to wide scale deployment. Tokens are physical devices that contain cryptographic information that is linked to particular user, that it is usually password protected. Physical tokens may not integrate into a user's lifestyle easily, and may infringe on the users behaviour. Some tokens may require specialised hardware and software to use, so limiting the mobility of such a solution. The cost of tokens as well as the cost of token management (to issue, maintain and revoke) further hinders widespread adoption [76].

A challenge for IAM solutions is to provide authentication methods with the strength to meet the growing threats [97]. As noted in the article, using passwords alone is not a strong authentication method. Even if the "best practices" for creating strong passwords are followed, it is still a single factor of authentication that is subject to dictionary and brute-force attacks. In order to make a password strong, it has to be crafted in such a way that it is very difficult to guess, but rather easy to remember. Using the various "Best Practices" heuristics will help in creating stronger security systems. To further the usefulness and strength of passwords, the authentication mechanism should allow for a small number of attempts before realising it might be an attack.

3.4.1 Tokens

Tokens are physical devices that has been specifically assigned to a person with a particular use in mind [21]. These are usually coupled with passwords in order to use them, increasing it to a two-factor authentication mechanism. These tokens contain cryptographic keys and can be represented in a number of schemes. It is also possible to put other constraints on the token, such as limiting its use to a specific time window. However much usefulness and security tokens provide, it is a double-edged sword. The loss or theft of a token is similar to the inconvenience and security risk of losing a set of car keys, that is, locking out oneself and allowing who ever finds the keys access to ones car. Consider a USB token, it is the size of a flash memory drive, and easily misplaced or left behind. It is a type of device that the user will only notice missing when the user next needs to make use of it.

The concept of *user centricism* is used to support a model, using a Personal Authentication Device (PAD) to tie together separate service provider-issued identity credentials [61]. The device is used to store separate notions of user identity and their related cryptographic keys and passwords such that the difficulty of credential management is eased. Furthermore, processes could be automated to the extent that, once the user authenticates them-self to the device, very little user input is required to access associated service provider-related resources. The device itself is able to make use of ubiquitous and mobile computing technologies, allowing it to integrate seamlessly with existing devices and systems, such as laptops, mobile phones and security devices. Since each SP must keep track of its population of users, it is able to make use of user specific PADs in order to establish a trust relationship. However, since the user uses the PAD to store user specific information and keys, it becomes a single point of failure, such that its loss or theft represents a serious breach in security. It is not clear from the model how this type of catastrophic failure is avoidable and recoverable.

In the financial sector, First National Bank (FNB) can offer its clients increased personal security for use in on-line transactions [35]. Users may incorporate a "one-time password" via SMS to their mobile phones. Users may also purchase a DigiTag, small handheld device, used to generate a random security code to authorise banking transactions. On-line users are required to combine an additional identifying factor with their normal account credentials for authentication.

PayPal, the on-line finance service, provides a similar security device to FNB that generates a random six-digit code every 30 seconds [86]. This code is incorporated into the log-in process. The purpose is to mitigate phishing attacks on the user, whereby users are tricked into disclosing log-in credentials to fallacious websites. However, more sophisticated phishing attacks that attempt to deprive users of the six-digit code and account details arise, where the information gleaned is utilised in breaching the user's account.

A detractor from widespread adoption of tokens is the difficulty in which a token may integrate into a user's lifestyle and habit. Tokens are small and can easily be lost or forgotten. Users tend to want things to be made easy; they are lazy.

3.4.2 Biometrics

The use of biometrics is a trade off between security it provides and the inconvenience it causes [71]. Biometric systems are more physically intrusive to a person using them while being able to increase security by decreasing the probability of identity theft. For instance, a retina scanner is usually in a fixed position requiring people of varying heights to make use of it and be subjected to an uncomfortable blast of light. A further concern is of privacy and control, where biometric systems that can match identities based on biometric information are reminiscent of "Big Brother", and represent real concerns for the public. Another concern is of the storage of such information, as people are left with no corrective measure should their biometric information be compromised. In [21], it gives the different biometrics as degrees of intrusiveness corresponding to effectiveness and reliability. Due to size and cost considerations, specific biometric approaches are only feasible in limited circumstances. The security employed is based on the cost of not securing something, that is, securing something that is more costly to lose than secure.

Among the methods of extracting biometric information fingerprints are considered the less intrusive [113]. However, this method suffers from inaccuracies in terms of acquiring a suitable sample, correctly matching samples with the corresponding identity, and dealing with errors. Fingerprint recognition is still weak when dealing with anomalous samples although fingerprint systems are considered the most accurate low-cost biometric [71].

The deployment of biometric devices are perhaps only feasible for organisations that require high levels of computer and physical security. Biometric devices are not suited to wide-scale deployment due to high costs of hardware, technical expertise and skills, and management. Furthermore, error rates within biometric systems can, in the worst case, compromise the security of the system or at the very least sour the user experience [31]. False accept rates are based on the probability that an unauthorised individual will be accepted as a valid member. False reject rates on the other hand describe the probability that a valid member will be rejected. However, biometric systems have their benefits: they reduce the reliance on passwords, reduce costs in terms of support personal (help-desk) and user management, and reduce fraud in terms of identity theft or impersonation.

3.4.3 A New Approach: Mobile Phone

The thought of using a cellular phone as a physical security token is not a new idea, and is rather obvious when any amount of thought is spent on multi-factor authentication in an open environment. Mobile phones are pervasive, with a high percentage population penetration in developed countries, and are becoming more important in developing countries. In 2006, South Africa had 66% of the population as mobile subscribers [44]. By the end of 2005, 7.5% (3.6 million people) of the population in South Africa had some form of Internet connection [125]. For many people in developing countries, mobile phones represent the only means to connect to the Internet. In fact, mobile phones represent the only connection to the outside world beside the dirt roads for many rural areas in Africa. It is obvious that GSM networks and other wireless technologies are feasible ways to connect out lying regions that are lacking in the most basic of amenities, namely running water and electricity. Even in urban areas, mobile phones are pervasive enough for the assertion that any person who connects to the Internet (via a fixed-line or broadband connection) has access to a cellular phone to hold true. The mobile phone is a device so entrenched in the urban dweller's daily routine that it is never far from its user. Take a moment to think, as a regular person, how soon you would notice the loss of a cell phone as opposed to the loss of a credit card sized smart card token.

RSA Security Mobile is a solution that generates a one-time password delivered to the user's mobile phone, pager or email address to be used to log in to access protected resources [104]. SIM Strong has a similar approach, where the mobile phone is used in conjunction with Liberty Alliance standards [118]. Here a user has the choice to interface the SIM directly with the PC via specialised connections and card reader hardware and software, or use the phone to send a SMS to the server. This approach also sends a session ID to the phone so that the user is required to correctly enter information into the browser.

Wu, in a PhD Thesis that focuses on counteracting and fighting phishing from the user interface, asserts that the one-time password is not suitable from a user point of view [126]. In that the one-time password is liable to be mistyped by the user, and poses usability questions for the system. Wu proposes a more usable approach, where the cell phone is used to approve a session so that a random session ID is generated and delivered to the phone. This also relies on the user to check the generated session ID against that which is displayed in the browser once the session has been approved. Another advantage of this approach is that the user will be alerted of an attack when a request for a session arrives at the user's cell phone that the user did not personally initiate.

Another approach using mobile phones is to make use of a "location factor" [76]. A mobile network operator can determine the physical location of a mobile phone. The accuracy of determining the location of a mobile phone depends on how dense the area is in terms of cells that make up the mobile network. It is proposed that authentication to certain systems can only be done when the user is in a specific location. For example, only by connecting the phone to the PC connected to a network can a user be authenticated to access the network.

A slightly different approach by Abdelhameed *et al.* [2] incorporates the use of a mobile phone in authenticating a user to a laptop. However, the model does not make use of the mobile phones GSM network, but rather connects to the laptop via Blue-Tooth. The system allows for Zero Interaction Authentication (ZIA) that strictly requires that the users mobile phone be within Blue-Tooth distance of the laptop. The mobile phone is used as a physical token that must be in possession of the user in order to access the laptop.

A similar approach to the framework presented in Chapter 5 is provided by MacDonald and Mitchell in [81]. Here the focus is on the use of a mobile station, comprising a mobile device and SIM card connected to a GPRS mobile network, as means of authenticating a user to a web service and enabling on-line transaction payment. This uses the mobile operator as a Trusted Third Party in conjunction with a Content Provider web service.

3.5 Chapter Summary

Privacy is shown to depend on how personal and confidential user information is stored and how it is disclosed. Privacy to the user represents control over their own information, in that they decide when to disclose that information. Pseudonyms can add a layer of protection by creating an alias through which users can transact with in the environment. Credential disclosure protocols, especially through trust negotiation, can control the gradual release of confidential information that results in mutual authentication. It further behooves the user to be aware of the privacy practices of an organisation they interact with as privacy polices should stipulate what information is captured and how it is processed.

The inability to distinguish the identity of a subject within a population is the basis for anonymity. This can protect individuals while providing a means for attackers to commit fraud. In that, mechanisms for providing and enforcing accountability should be available. Anonymity finds place within the communication channel, obfuscating traffic such that observers are unable to link a message with its participants. Two common approaches are the use of proxies and trusted third parties.

Approaches to multi-factor authentication find that biometrics are considered invasive and may compromise the privacy of individuals. Furthermore, physical tokens are desirable if they do not impact on the user experience. The main purpose of Section 3.4.3 is to expose the properties of the mobile phone as an ideal security token. The following chapter discusses two abstracted models for achieving privacy and anonymity.

Chapter 4

Achieving Privacy and Anonymity

4.1 Chapter Overview

In the previous chapter we discussed the concepts of privacy and anonymity. Although closely related, these separate concepts involved different approaches. Privacy is shown to depend on the nature of the storage and disclosure of credentials and personal information. A concern is that once information is divulged in a networked environment one has no or little control over how that information is used or disseminated. Anonymity, however, is the protection of one's identity by keeping it secret or hidden while in communication. This is not an easy feat within a networked environment as one leaves traces of one's activities. This chapter presents two models; a model to provide anonymity for participants in communication within an open environment, and a second model to address issues of privacy.

Section 4.2 presents an abstract model that describes an approach to anonymise traffic over the Internet. It does so by describing an Identity Agnostic layer which is responsible for obfuscating the communication channel and traffic. While anonymity is ideal for the user there must be balance for the administration of the domain in terms of accountability. The actual model is described in detail in Section 4.2.1.

In an effort to provide privacy as well endow the user with control over their personal information a model for privacy is presented in Section 4.3. The idea is to unify the accounts from different domains that belong to the user into a meta-identity scheme. The model is formulated in Section 4.3.1. The chapter summary is provided in Section 4.4.

4.2 Towards an Identity Agnostic Layer and Anonymity

The physical Internet, as discussed in Woodcock [124], is a global conglomerate of smaller networks that are able to communicate and transfer information amongst themselves. These networks are arranged and identified by a hierarchical structure of domains called the Domain Name System (DNS). The DNS is able to describe and assign unique names to individual networks, allowing for the growth of the Internet. Within each domain there exist machines, such as servers, gateways, and routers, that are responsible for network administration and management, acting as controllers of the underlying physical network.

The conceptual or logical view of the Internet is such that it can be viewed as a cloud, made up of machines, end users, and particular services. This allows virtual layers to arise, regardless of physical properties and location. Upon this abstractions can be made, as illustrated by Figure 4.1. One such abstraction is an Identity Agnostic (IA) layer.

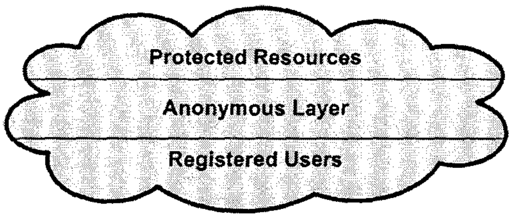

Figure 4.1: A conceptual view of the Internet

Within the Internet there are protected resources that require registered users to go through some authentication process to access these resources. *Registered* is a loose term for describing a user who has completed some form of registration with an identity-issuing authority. The registration process encompass the broad range of requirements and restrictions that can be imposed on a user. For instance, it is case of website membership that a user be registered to make use of that website. *Protected* resources are digital resources that are secured against free and general access. Securing a resource entails creating processes that control access to the resource in such a way that a user has to be authorised and authenticated before access is granted. For example, it is required of an *Ebay.com* user that they log in by supplying their user-name and password before engaging in any bidding activity. Figure 4.1 also depicts an anonymous layer, describing users and resources whose communication and access is regarded as open. To clarify, if a resource is not protected and allows unfettered access within the bounds of normal physical and logical constraints, such as bandwidth quota or the number of connections, it can be considered a part of the anonymous layer. Users whose identity do not matter, beyond the normal characteristics and properties attributed to a host, such as IP address and OS/Browser information, can be conceptualised as being within the anonymous layer. Although the word *anonymous* is used, it is inaccurate to describe the entire communication transaction as anonymous, as the participants of the transaction, and even the contents of the transaction may be discovered. Rather, the usage is in the sense that the identity of the user is of no consequence in that user's pursuit of available resources. This is typical of general web browsing.

The registration of users can be used as a means of controlling bandwidth and traffic between a content provider and users. The content is still freely available, however, its access is controlled by the fact that the user has to have a registered account. Once authenticated to the content provider, the provider is able to control the amount of data downloaded from the site by that particular user. It is still possible to control bandwidth usage, without requiring user registration, by incorporating download scripts that can throttle the download capabilities of a host. By making resources protected and incorporating the registration of users to access those resources, then greater control is generated by the domain or service provider.

The Basis for an Identity Agnostic Layer

If the Internet can have its constituents categorised as *registered*, *protected* or *anonymous*, then it can be put forth that there can be a layer that embodies all three. An Identity Agnostic layer can be described as a virtual layer of the Internet that allows quasi-anonymous activities to take place. The qualifier "quasi" is employed to denote that accountability is not forsaken. But rather accountability is a core concern, where the IA layer is a balance between user anonymity (protecting the user) and accountability (protecting the domain).

In Section 3.3 we covered the concept of anonymity, showing how anonymity can be used to protect the user. Specifically, observing the channel can reveal information that can empower the attacker to orchestrate more sophisticated attacks against the system or users [67]. Thus from the perspective of the user it is better to have their identity obscured and their traffic secured against observation or modification.

From the perspective of the organisation, which has financial interest in the correct operation of their networks and systems, it is better to have confidentiality and accountability so as to protect the domain. Accountability entails the enforcement of decisions made pertaining to access control and authorisation, whereby actions are attached to a particular user. This allows any monitored breach in policy or system security to be followed up and rectified. Rules without associated negative consequences for transgressions are meaningless. In other words, not enforcing a rule makes the rule redundant. For accountability, it is necessary that the system logs each particular action or event. Organisations have other reasons to have the logging of transactions, from legal obligations to meeting quality standards [18]. Furthermore, it is supported by Borcea-Pfitzmann [18] that identity management systems should assist users by logging transactions and data transfers they engage in.

4.2.1 Model for Anonymity

In this section we present an abstract model designed to achieve an Identity Agnostic layer that balances anonymity and accountability. In order to provide anonymity one needs to obfuscate the message and the channel between communicating parties. Obviously this entails making the contents of the message secret and confidential, typically by way of encryption. But this does not ensure that the address of the sender and receiver remain anonymous. Furthermore, in an open environment one cannot control the communication channel, so attackers may intercept messages and be able to identify the sender and receiver. An perceptible approach to obfuscating the channel is not allow direct connections between the parties, but rather create a complex route in which a message's final destination is not evident. Specifically, in routing the message, the original sender and ultimate destination remain secret whilst only the intermediate hops are visible. From this an attacker may not infer the identity of the sender and receiver unless the attacker controls the intermediate routing points.

Figure 4.2 represents a model that describes the components that can be used to create an IA layer within the Internet. The user is protected by having the communication channel and the domain environment operate in such a way that it attempts to decouple the transactions of the user from the system. This is achieved by having two separate end points that exist within the Internet that represent secure point-to-point connection to participating parties. A participating party in this regard is a generic host or user, and a generic organisation domain or web service. As such, the participating parties, via their own particular connection to the Internet, connect to a secure anonymous server. The Secure Anonymous Server (SAS) is a machine that acts as a portal to the identity agnostic layer.

Obfuscating the communication channel, as depicted in the diagram, means creating a virtual layer that operates between the two secure portal end-points. Approaches to providing this level of anonymity are covered in Section 3.3.1. However, it is worth noting that all the anonymous routing approaches covered in the afore-mentioned section requires the assistance of a Trusted

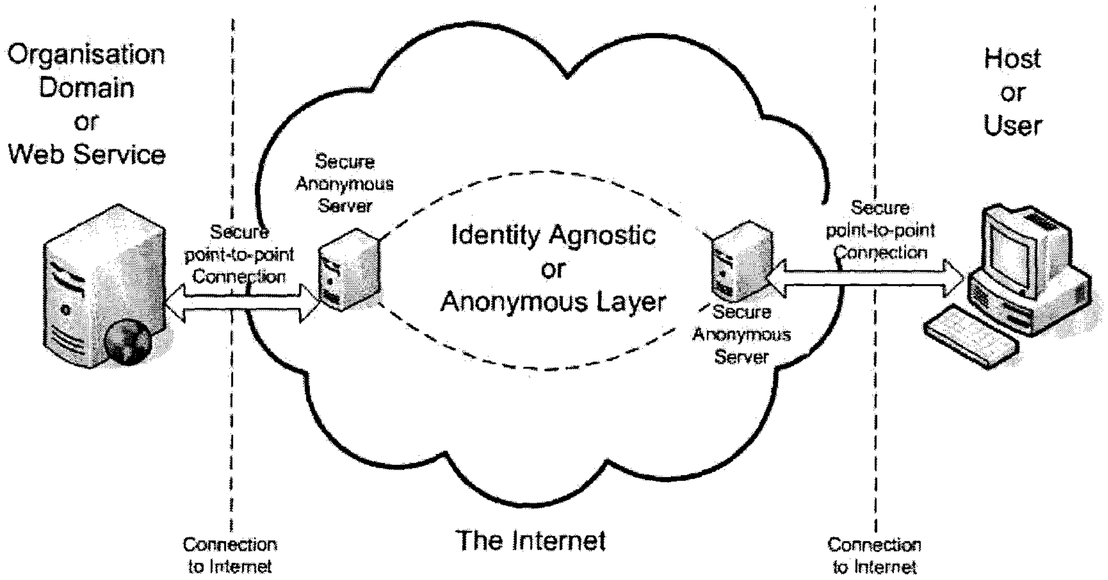
Organisation Domain or Web Service
Secure point-to-point Connection
Secure Anonymous Server
Identity Agnostic or Anonymous Layer
Secure Anonymous Server
Secure point-to-point Connection
Host or User
Connection to Internet
The Internet
Connection to Internet

Figure 4.2: An overview of the Identity Agnostic Layer

Third Party (TTP). Essentially, the common theme among the approaches is to expose a proxy machine that initiates the communication stream between a sender and the intended receiver. Additionally, a series of distributed machines perform the routing between the proxy end-points. This is observed in both Onion Routing [94] and MIX [90]. The use of TTPs achieves the anonymity of both the sender and receiver [67].

In order to assure accountability, extensive and accurate logging of transactions must occur. Such an activity has no direct bearing on the operation of the layer and is not included in the diagram for that reason. For all intents and purposes, consider the logging to be a hidden action.

Secure Anonymous Server

The SAS in Figure 4.2, although forming a visible point of attack, can provide authentication through the process of establishing the secure point-to-point connection or by means of multi-factor authentication. The SAS must be able to maintain connection state in order to act as a proxy, as it routes incoming and outgoing traffic within the layer. However, the user and other participating parties must be registered with the layer. Identification of users should be based on temporary pseudonyms, used in decoupling user ID and user account. The SAS connected to the client should be aware of the link between pseudonym and user account by way of connection rather than a look-up table approach. This forms the accountability part of the approach, where logged transactions are linked to particular entities within the system. It is necessary that this information be hidden from the public network, and be a part of the anonymous layer.

Identity Agnostic Layer

In order to attain an anonymous layer, the SAS represent entry and exit points to and from the layer. The layer, depicted in Figure 4.2, itself is made up of a peer-to-peer architecture where each node is a SAS. The reason for this design consideration is that each SAS can act as an entry or exit point for the layer, and that each node is a router. Thus increasing the complexity of the layer, and not limiting the potential of the entry and exit points.

Communication between two clients is handled by the initiating client establishing a connection with a particular SAS, where another connection is created from some SAS to the indicated destination. Once entry to the layer has been gained, a random path routing mechanism is engaged. For each packet, in between the two designated SAS, a new path over a minimum number of hops within the layer is established.

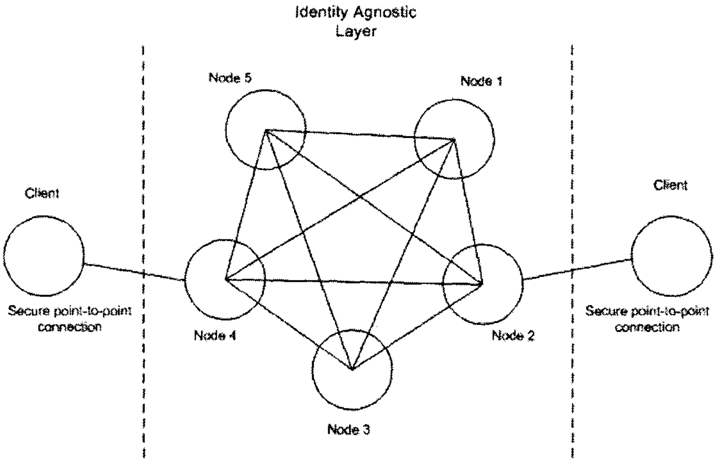

Figure 4.3: Peer-to-peer Identity Agnostic Layer

Figure 4.3 depicts the peer-to-peer architecture of the anonymous layer, where each node is a SAS and the distance between each node is a single hop. A message sent to the layer should obfuscate the final destination, as well as the original sender. This is to protect the identity and thus ensure privacy for the user. The traffic within the peer-to-peer architecture should necessarily be encrypted [21]. This is to ensure the confidentiality of the message contents, as well as protecting the protocol of the layer.

Clients

In order to connect to the Secure Anonymous Servers, some form of client application is required. This will handle client side authentication with the SAS. Hosts and organisation domains represent clients in this model. Clients will connect to a SAS using a secure point-to-point technology, such as SSL/TLS or IPSec [38], providing encryption of the traffic. This will protect the communication channel between the host and SAS from being easily observed by a third party. At most, an observer will be aware of a connection yet incapable of discerning the nature of the connection. However, a connection can be instantiated from either direction, as such, a mutual authentication process should be incorporated [85].

Transaction Logging

A transaction log is a record of actions taken by users or systems, describing the characteristics of the action. The point is to track each message and connection created, capturing the source and destination node addresses, the time information, and the message size. Furthermore, it is possible to perform analysis of the logs such that non-obvious information may arise. For example, it might reveal weaknesses in the algorithm that handles the routing decisions, or even an alert that a particular node has anomalous behaviour.

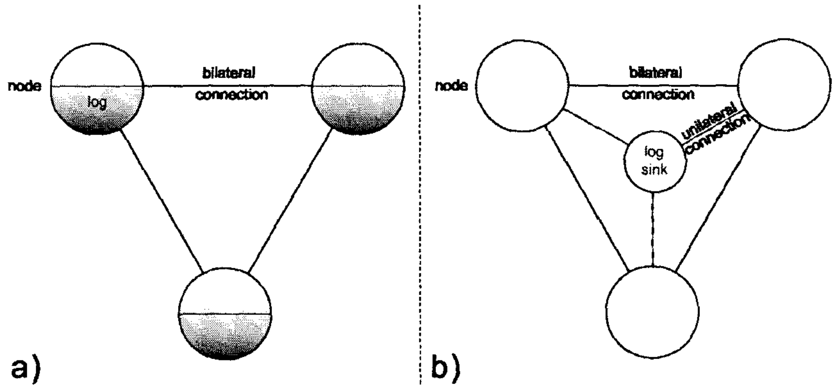

Figure 4.4: Transaction Logging: a) Node Logging b) Log Sink

The literature does not seem to provide for a secure means of logging transactions without the risk of compromise or subversion. However, a machine that engages in logging can be secured in the same manner as other machines so as to reduce the risk of being compromised. There are two options available in this model, as illustrated by Figure 4.4. Essentially these can be described as either a decentralised or centralised model. Node logging, the decentralised approach, specifies that each node creates and stores its own transaction logs. This suited to the peer-to-peer topology of the anonymous layer.

In the centralised approach, logs are transmitted from each node to a central location. A log sink in this model is a hidden node which has unidirectional connections from each node in the anonymous layer. Because traffic is transmitted in one direction, the log sink can be made to accept a certain type of connection. In that, the log sink will not be responding to unnecessary connections. Additionally the node will not send information out. This has the feature of reducing the chances that an attacker may compromise the machine. It is also easier and more efficient to perform analysis on logs that are stored in a central location. The disadvantage of this approach is the increased use of bandwidth and the inability to discern the status of the log sink.

4.3 Privacy Through The Meta-Identity Portal

This section presents a model that attempts to provide the user with privacy and control over their own credentials. Privacy is a primary goal of IAM systems, in that confidential user information is protected from being used against the user and the system. Anonymity is also the protection of the users by removing identifiable features from an environment that camouflage a user within a set of users. A technique to achieve a level of privacy is the use of pseudonyms, where the link between a pseudonym and the user's identity/account is not public knowledge and should remain hidden to be effective [4, 29]. To provide a level of anonymity, it is required to then proxy the account access and to use pseudonyms. Effectively any attacker observing the channel and the flow of information between user and the IAM will not be able to discern the links between the pseudonym of the user's account and the actions of that user.

4.3.1 Model For Privacy

Privacy is achieved by the fact that a user's credentials will be used as little as possible, and control over user credentials will be resting with the user. There are of course, some accounts that will require a user's full credentials, where a pseudonym will not be acceptable. For example, a bank will require a customer's full set of details, where as a community forum typically only requires a valid email address. The portal will have a master account, which represents the user's real world identity, and upon which all the other client accounts are based. A user's credentials will be stored in the master account. Existing accounts on other Identity Providers (IdP) will be managed from the portal. New accounts will be created using derived forms of user credentials, in essence account details will be pseudonyms themselves. In that the new account will have no discernable identifiable features of the user's real identity, unless of course, the user would like to disclose such information. Client IdPs will be categorised by whether they will accept pseudonymic credentials or require real credentials.

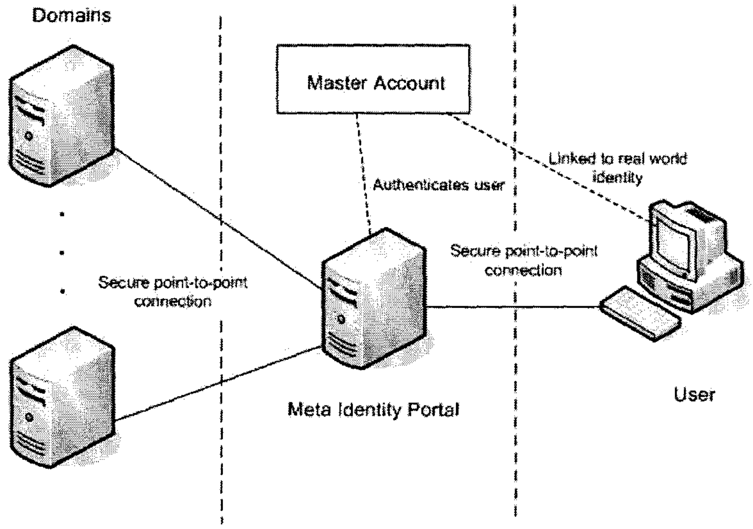
Domains
Master Account
Authenticates user
Linked to real world identity
Secure point-to-point connection
Secure point-to-point connection
Meta Identity Portal
User

Figure 4.5: The Meta-Identity Portal Overview

Figure 4.5 shows the overview of the model where the master account is used by the user and the meta identity portal to authenticate the user to accounts within various domains.

4.3.2 Identity and Service Providers: Accounts Of The Domains

In this context, a domain is made up of Identity and Service Providers, which are independently operated and have their own agenda and requirements. Figure 4.6 gives an example of a domain, indicating the relationship between the user who has an account with an Identity Provider (IdP) that is connected to several Service Providers (SP), all of whom control access to resources. This is a typical model of federated identity, for more specifics refer to Section 2.2.

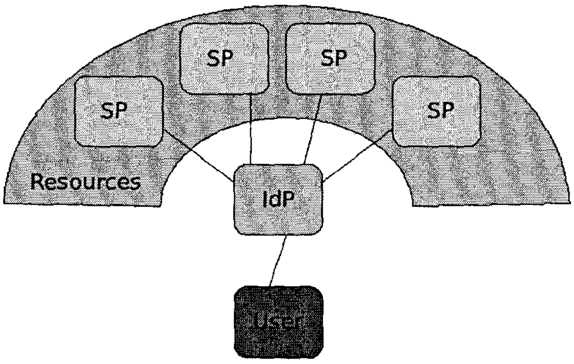

Figure 4.6: An example of a Domain

Identity Provider

An IdP is a system that stores user information (credentials and attributes) in an account. Account management is accessible through the authentication process. An account is created when a user discloses the required information and meets the conditions for joining the IdP. A typical IdP has a front end which authenticates a user when the user logs in, or authenticates a user against information supplied via the SP. The IdP has a back end that stores the user information, and logs transactions for accountability and quality assurance reasons. An IdP can be connected to several SPs in a "Web of Trust" [16].

Service Provider

A SP provides access control to protected resources and services that authorises access to authenticated users. Protected resources can be any digital resource that can be accessed or transferred in a networked environment. Generally, it is wise to separate your IdP from your SP so that should one be compromised, it will not necessarily mean the other will be compromised.

Derived Account

A user account is created and controlled within an IdP, which is basically a record of credentials and attributes that describe that user. This means that personal information of a user specifically required by an IdP is stored there. Should the identity store be compromised, the privacy of users is also compromised. It is assumed that a compromised identity store is one such that the security measures in place have been defeated or bypassed. A further consideration is that a user may have several accounts with separate domains, such that there is duplication of personal information. In that, the same or similar sets of personal identity information belonging to a specific user is stored in several different locations under separate pretexts. An individual concerned about privacy and aware of the dangers may hesitate to disclose the same information to different domains, but may want to store personal information in a single place. This is the basis for the "derived account".

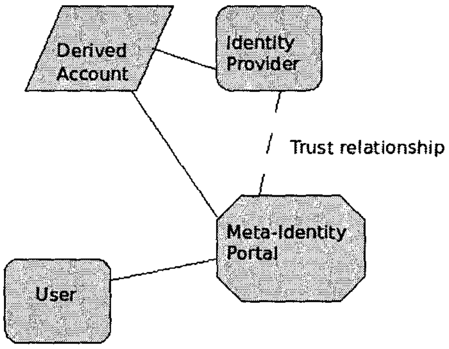

Figure 4.7: An example of a Derived Account

A derived account, as shown in Figure 4.7, is where the credentials and attributes are not stored directly with the IdP but are based on a "master account". A master account, described in detail in the next section, is a single account that contains the user's personal identity information. For each derived account, each credential and attribute is a cryptographic derivative of the master account. This means that an account specific to a domain will not contain identifying information where the only link is a pseudonym or a cryptographic reference. This, however, relies on the nature of the trust relationship between the meta-identity portal and the IdP.

4.3.3 The Master Account

A master account in the context of Figure 4.5 represents the base or root account on which all other accounts are derived. The derived accounts represent the nominal accounts managed by IdPs. In understanding the nature of the master account consider Section 2.2.5, which deals with Public Key Infrastructures (PKI). An aspect of a PKI is that the certifying authority is in effect a Trusted Third Party (TTP). This means that the participants and users essentially trust in the correct operation of PKI in order to continue with their business. The master account contained within the meta-identity portal acts in the same way that a certifying authority does; it allows everyone to trust a single entity rather than having to manage a complex web of trust. However, the implementation of such a scheme may not necessarily employ the same technology.

The purpose of the master account is to store credentials and attributes that may or may not be certified. The decision to have personal information certified depends on the nature of the derived accounts the user intends to use. Certifying personal information involves an individual having their real world identity confirmed by a designated certifying officer [14, 122]. This entails providing physical proof of their identity in person, such as presenting an Identity Book or passport.

4.3.4 Meta-Identity Portal

The meta-identity portal, as illustrated in Figure 4.5, acts as a proxy between the user and domains specific to the user. This is similar to meta-identity systems that tries to unify user accounts in the sense that different user accounts are accessible from the same location. Meta-identity is the convergence of control of multiple disparate accounts to a single point, this is discussed in further detail in Section 2.2.4. It also represents a single system to which the user must authenticate in order make use of the myriad accounts under the user's ownership. A feature and benefit of such an approach is Single Sign-On. Additionally, the meta-identity scheme fosters a *user centric* environment.

The user engages in a process that requires that user authenticates against their master account. Once authenticated the user is free to proceed in a manner of their choosing. Access to services in different domains is proxied against the master account, without the user being required to divulge private information. Since the portal acts as an intermediary, it becomes a trusted third party.

The portal itself should allow connections through secure point-to-point technologies. For example, SSL/TLS or IPSec [38]. This has the feature of mutual authentication of both the portal and the user. Furthermore, the communication traffic is kept confidential, as it is encrypted.

4.4 Chapter Summary

In this chapter, two separate models are presented in an attempt to address issues of privacy and anonymity. The model for anonymity is such that it obfuscates or hides the identities of parties in communication. This is achieved by postulating a virtual layer called an Identity Agnostic layer, where it is posited that it is possible to achieve anonymous communication in an open environment while still assuring a level of accountability.

The second model, a model for privacy, is achieved by incorporating the meta-identity approach of unifying the control of disparate user accounts to a single location. This follows the *use*r c*entric* paradigm, giving control of personal information to the user. Single Sign-On, as a side benefit, may enhance the user experience. Through the use of a master account and pseudonyms other accounts are designed as cryptographic derivations. The disclosure of personal information is then at the discretion of the user.

Chapter 5 attempts to address the lack of a pervasive and ubiquitous solution to multi-factor authentication. The solution is presented as a generic framework in an effort to follow the tenets of de-perimeterisation.

Chapter 5

Mobile Phone Authentication Framework Solution

5.1 Chapter Overview

Chapter 4 covered privacy and anonymity, presenting abstract models to remedy those issues identified in Chapter 3. This chapter aims to present a generic framework that addresses the lack of a pervasive and ubiquitous solution to multi-factor authentication. The generic approach attempts to broaden the applicability of such a framework, rather than as a solution to a specific problem domain. This approach makes use of mobile phones, mobile operators and their GSM network.

The layout of the chapter is: a discussion on the security aspects of mobile devices and GSM networks in Section 5.2. This is to briefly detail the security environment in which these entities exist, such as the mechanisms in place and their possible shortcomings. However, since these considerations are typically outside of the control of the framework, they are discussed for completeness.

Section 5.3 details the environment in which the user exists, introducing aspects of the framework that are relevant to the user. These are the requirements of the user in terms of the components the user has, and clarifies specifically what the user is.

Section 5.4 encompasses the functionality of the mobile phone and SIM applications that make up the framework. The considerations of the mobile phone environment are covered in Section 5.2.1.

The role of the mobile operator within the framework is detailed in Section 5.5. These cover the operation of the different servers and their interfaces that comprise the mobile operator environment.

Section 5.6 briefly covers the web of trust domain environment, specifically the web service that provides an interface between the mobile operator and the user.

Once all the environments have been introduced and explained, Section 5.7 describes the interactions between these environments and under what circumstances communication takes place. Finally, the chapter concludes with Section 5.8.

5.2 Security Considerations

Since this framework incorporates mobile devices that make use of GSM networks, it is relevant to briefly discuss the security considerations of these entities. In this section we cover the security environement of the mobile equipment and SIM. A further discussion concerns the security over the GSM network.

5.2.1 Mobile Phone and SIM Security

Mobile devices, such as phones and personal digital assistants, are computing platforms in their own right. This is due to the fact that they have hardware components similar to other computing platforms. In other words, mobile devices have processing and memory capabilities, as well as various operating systems that allow applications to make use of these resources [45]. The mobile phone as computing platform is then as fallible as other platforms, and relies on the OS to secure the environment. In response, there are anti-virus applications for mobile devices that aim to protect the device from malware.

However, there is a separation from the mobile phone and the SIM of a phone. A mobile phone is comprised of the Mobile Equipment (ME), used for GSM protocol communication, and the Subscriber Identity Module (SIM), used for security related functions, such as cryptographic functions [111]. Communication between ME applications and the SIM application is handled by the SIM Application Toolkit (SAT) [1], a standard that provides mechanisms which allow applications, existing in the SIM, to interact and operate with any ME which supports the specific mechanism(s) required by the application. The SAT allows for operators to create specific applications that will reside on the SIM, with a means for interaction between the ME and SIM [117].

Since the operators own the SIM , they can use these applications to provide operator-specific services to subscribers. An added advantage is that implementations of SAT are GSM network technology independent, such that it can blend seamlessly between 3G and 2G networks.

An alternative to employing SAT as a means to enable application communication over GSM is to use Unstructured Supplementary Services Data (USSD). It is a session-oriented technology transmitting data as clear text. However, USSD suffers from attacks on the GSM backbone, where, with the appropriate tools, it is possible to intercept and modify messages [117]. Although SAT also suffers somewhat due to interoperability issues between different SIM vendors, SAT remains the more secure technology, which is naturally structured in a traditional server-client architecture.

5.2.2 GSM Security

In Kasera and Narang [63], network access security is used to encompass GSM network and "general security over the air" concepts. This includes four important features:

- Mutual Authentication
- Data Confidentiality
- Data Integrity
- User Identity Confidentiality

Mutual authentication entails the serving network verifying the identity of the subscriber in the user authentication process, and the subscriber's network authorises the serving network. In some cases, the serving and subscriber networks are the same [54], however, a serving network may be an intermediary network. Data confidentiality is the encryption of the contents of a message being transmitted over the radio network. Data integrity is the process by which the message is verified that it is unchanged. This is based on encryption keys established at the time of connection and using it to create a 32-bit Message Authentication Code (MAC) which is appended to the end of the message. The received MAC is compared with the computed MAC of the received message on the receiving end of the communication stream. User identity confidentiality is achieved by creating a temporary pseudonym to be used on the radio link of the network. This is handled by the local location area where the user is registered. This is based on the Home Location Register (HLR) value for the user, which tracks the global location of the user [63].

GPRS uses a new ciphering algorithm optimised for packet data transmission [89], but uses the same authentication mechanisms as GSM. Gateways are provided by the operator that are able to route messages over GSM using 3G protocols to and from IP-switched networks [127]. The payload of such messages are encrypted by normal means of the mobile equipment, though this does not carry over to the IP network, unless of course, the IP payload itself is encrypted.

In short, GSM security is established such that a mobile application does not need to know about the protocols to affect the process.

5.3 The User Environment

This section introduces the different components of the framework and their respective environments. A user is described as an end user of services, and a subscriber is a person with mobile phone and has an account with a mobile operator. This framework details how an average user and subscriber can make use of the mobile phone as a second factor in authentication within a Web of Trust (WoT) domain. A WoT is a group of organisations that form a single security domain using Federated Identity technologies.

5.3.1 The User

A user has a mobile phone and is a subscriber to a mobile operator. A user also has a workstation, a computing device that has some form of a connection to the Internet. A user has services and resources that they will want to access, as well as information to be protected.

5.3.2 The Workstation

A workstation in this context is defined as any device that is capable of computing, digital data storage and provides the same functionality as a PC. For example, a desktop PC, laptop or a PDA. Furthermore, this workstation needs to have a connection to the Internet and a Web Browser.

5.3.3 The Browser

A browser can be viewed as a communication tool that represents hardware and operating system independent lightweight client software that bypasses inter-operability concerns. Using a web browser to connect to a web server ignores the underlying operating system of both the client and server machines. The security of the browser environment cannot be greater than the security of the underlying operating system, where the security of the browser protocols cannot be greater than the security of the browser environment [93]. Although it is outside of the scope of the discussion, the security of the underlying operating system is a concern, where a compromised operating system allows the possibility for input and data from the user to be captured.

Within the context of the framework, the browser is the client software that provides access to protected resources over the Internet to the user. Here, cookies are used for temporary information persistence, maintaining connection and user state information. The WoT domain identity and access control-related protocols are HTTP-based, meaning the browser does not need specialised software.

5.4 The Mobile Phone Environment

This section is to provide a conceptual view of the GSM token web service client for the mobile phone, exploring the functional requirements of the client. Together, the ME and the SIM of a phone comprise the subscriber client for the GSM token web service. However, in order to be implementation agnostic, the logic and the user interface are separated. In other words, the phone specific component on the phone is restricted to the User Interface (UI). A SIM application performs the program logic. The SIM is protected by a PIN (Personal Identification Number), while the ME application is password protected.

The SAT is suited to user-oriented services of a *request - response* nature, while also offering opportunity for server-initiated transactions. SAT is ideal for the approach of this framework because the mobile application has to respond from both user initiated input as well as from the mobile operator side.

This is demonstrated in (1) and (2) in Figure 5.1. A *Proactive* command is the mechanism for the SIM to affect changes in the ME environment, while the *Responsive* command reacts to calls from the ME. The GSM communication interface of the ME is also controlled through

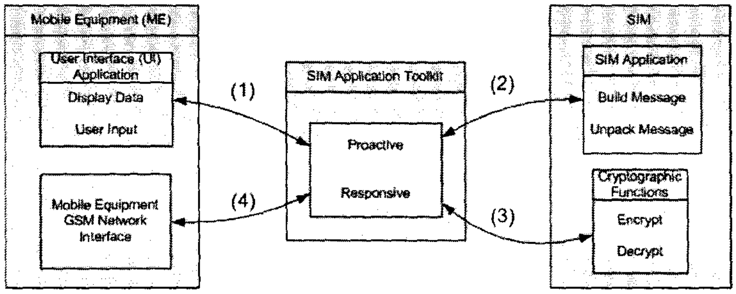

Figure 5.1: The ME and SIM conceptual view

Proactive commands given as (3) and (4), where all GSM network communication utilises the cryptographic functions of the SIM.

5.4.1 SIM Application

The SIM of a mobile phone controls access to the network, requiring the subscriber to input a PIN to authenticate the user and the phone, using a valid IMEI (International Mobile Equipment Identity) number, to the GSM network [64]. The SIM operating system controls read and write operations, presenting a limited view to the mobile phone application layer, via the SIM Application Toolkit [1].

The SIM application will do as much of the logic processing as possible, in order to be as implementation and phone hardware independent as possible. If the ME application is used for information display, storage and user interaction, then the SIM application is used for building and unpacking messages, making use of cryptographic functions of the SIM, and interfacing with ME GSM communication module. Figure 5.1 shows an overview of the mobile environment, and the functionality of the SIM application. The SIM application, for outgoing communication, is responsible for building the message in response to user input. The message is encrypted using the features provided by the SIM, and transmitted to the GSM network via the SAT using the ME GSM network interface module. Unpacking messages involves firstly decrypting the GSM message and then extracting the message contents, displaying the relevant information to the user. Table 5.1 gives a list of message types and their description.

The ME application responds to the user input, as well as input from the GSM Token Web Service. The SIM application will run in the background until accessed via the ME application or an Alert notification is received. An Alert is when the mobile operator server receives a request for session from the WoT domain, in order to notify the user, the server sends an appropriate

Direction	Message Type	Description
In	Token Receipt	Receives the GSM Token
In	Alert Receipt	Receives an Alert for a session, generated by the WoT domain concerned
Out	Token Request	Creates and sends a request for a session
Out	Alert Response	Sends an indication of the action to be taken for a particular Alert
Out	End Session	Indicates which session to end

Table 5.1: GSM Token service messages

message to ME/SIM application which then alerts the user. The display is updated with the relevant information, and the user can start interacting with the application.

5.4.2 Mobile Phone Application

The ME contains an application space where the phone specific user interface is run. The ME also houses the mobile phone communication interface that is used for connecting to the GSM network. The purpose of this application is to secure the access of the GSM token service with a clear, concise and unambiguous user interface. Access to the client application features secured through the use of a password, which the user enters upon start-up of the application. The ME application is phone-specific and is predominately used for the User Interface (UI), which displays relevant information to the user and captures input from the user.

The ME application will display the list of currently active sessions, presenting information concerning each session, with options for the session to be terminated. The UI will display the list of unread Alert notifications, presenting information concerning each alert, and allowing the user to respond to the alert. The user can request a new session by choosing from a subset of parameters used to describe the nature of the request. Parametric information occurs as the intended WoT domain, the length of the session, and other implementation-specific criteria. The user should be able to decide the mechanism to be used in connecting to the PC.

Further features should be considered at the time of implementation, such as detailed user information feedback on operations, value-added services, advertisements, and new services. However, such considerations are implementation specific, whereas the focus is on the core functional aspects of the mobile phone two factor authentication scheme.

5.5 The Mobile Operator Environment

The mobile operator web service provides two different interfaces. The first is a connection to the GSM network to interact with the mobile phone. The other is a non-public interface over a public network for the different Web of Trust domains to verify a security token. Together, the following sections describe the means for providing a scalable and generic two-factor authentication scheme.

5.5.1 Token Request and Issue Service - TRIS

This interface is provided over the GSM network, such that it is only accessible from a valid subscriber whose phone is enabled to make use of such a service. The purpose of the web service is to provide a challenge-response mechanism during the authentication process. A request for a GSM Token can be initiated in two ways: Firstly, a mobile subscriber can initiate a request when that user intends to make use of services that require a GSM Token. The user of the mobile phone accesses the service via an application on the mobile phone which makes use of a protected SIM feature. The SIM on the mobile phone is PIN protected, where the subscriber is prompted when the phone is turned on. The mobile application is password protected, which the user must supply when the application is started.

Secondly, a Web of Trust domain can initiate the request procedure when the user tries to access protected resources. The web service sends an alert, describing the nature of the request, to the subscriber's mobile phone, where the subscriber can decide to authorise the request or not. With this feature it is possible for the subscriber to be alerted to unauthorised attempts to access the user's account.

There are two cryptographic keys used for a token; the *PseudonomKey* is derived as cryptographic key that uniquely identifies a session for a particular subscriber, and the *TokenKey* is created to encrypt the body of token. When a token is created, a Subscriber (*PseudonomKey*, *TokenKey*) tuple is added to the list of current sessions, where the *Subscriber* value identifies a unique subscriber.

5.5.2 Token Validation Service - TVS

When a user tries to access protected resources in a particular Web of Trust domain, the user may be required to present a GSM Token. The Token Verification Service (TVS) is used by the

Web of Trust domain to validate the GSM Token presented by the user. This can be as simple as returning a "yes" or "no" answer, though further assertions about the token and session can be made. When the TVS receives a token from the WoT domain web service, it decrypts the token using the *TokenKey* in order to verify that the contents of the token have not been tampered with. The TVS then checks the validity period of the token, given as the *ValidityPeriod* field of the token, such that the session is either approved or refused.

The mobile operator environment will necessarily track each active session, each outstanding request and alert. This is so that user-related actions can be logged to provide accountability.

5.5.3 GSM Token Structure and Validity

The point of cryptography is to make the cost of the work to retrieve the secret information greater than the value of the information itself [38]. A token can be viewed as a cryptographic container for relevant authentication and authorisation information. Even though a digital token is encrypted, it is important not to disclose any potentially confidential information that, should it become public, could potentially compromise the system or the user.

The TRIS creates a token to issue with two major fields, as shown in Table 5.2; the header is concerned with communication, and the encrypted body is to protect token specific information. The header contains a *WebServiceHandle* field that points to the address of the WoT domain web service.

The body is encrypted in order to protect the contents of the token against tampering. The body has a *TokenID* field, uniquely identifying this particular token. The GSM Token will record the date and time of the explicit range in time for which the token is valid, particularly the time between when the token was issued and when the token will expire, given by the *ValidityPeriod* field. The *PseudonomKey* field describes the temporary key assigned to the session, which is used on the mobile operator side to identify the session requested by the subscriber.

5.6 The IAM Environment

A Web of Trust domain can be viewed as a single security domain in which authentication and authorisation of federated users occurs. A Service Provider (SP) provides access control to protected resources that authorises access to authenticated users. Protected resources can be any

Message	Field Name	Description
Header	WebServiceHandle	The Address of the relevant web service
Body	TokenID	The identification number of this token
	ValidityPeriod	The explicit range for which the token is valid, given by date/time
	PsuedonomKey	The psuedonymic cryptographic key identifying the session

Table 5.2: Structure of the GSM Token

digital resource that can be accessed or transferred in a networked environment. An Identity Provider (IdP) is a system that stores user information (credentials and attributes) in an account. A typical IdP has a front end which authenticates a user when the user logs in, or authenticates a user against information supplied via the SP. Typically; users are authenticated when they present a user-name and password. The IdP has a back end that stores the user information, and logs transactions for accountability and quality assurance reasons. An IdP can be connected to several SPs in a Web of Trust.

5.6.1 WoT Domain Token Service - DTS

For a generic framework, the structure of the WoT domain and the communication protocols employed should be viewed from a high level of abstraction. An examination at depth will yield implementation-specific considerations, detracting from the purpose of the framework. It is important to note that web services are described at a high level of abstraction, where implementation goals can be achieved through several different approaches. With modular design, independent components, based on open standards, can be integrated into the existing system, providing extended functionality [41]. Such that, a web service in the WoT domain the can provide two interfaces, one with reference to the autonomous token transport mechanism described in Section 5.7.3, and the mobile operator specific interactions in Section 5.7.1. The purpose of the first interface is to transport the GSM Token to the WoT domain in a secure and confidential manner. The second interface is there to connect to the mobile operator Token Verification Service, which will check the validity of the token.

5.7 Interactions

This section describes the interactions between the major components, as illustrated in Figure 5.2. Section 5.7.1 describes the nature of the interactions within the framework between mobile phone specific applications and the TRIS. Section 5.7.2 details the connectivity between the PC and mobile phone. Section 5.7.3 describes the communication process between the user and the Web of Trust. Finally, Section 5.7.4 describes the back channel connection between the WoT domain and the TVS, as well as the interaction with the TRIS.

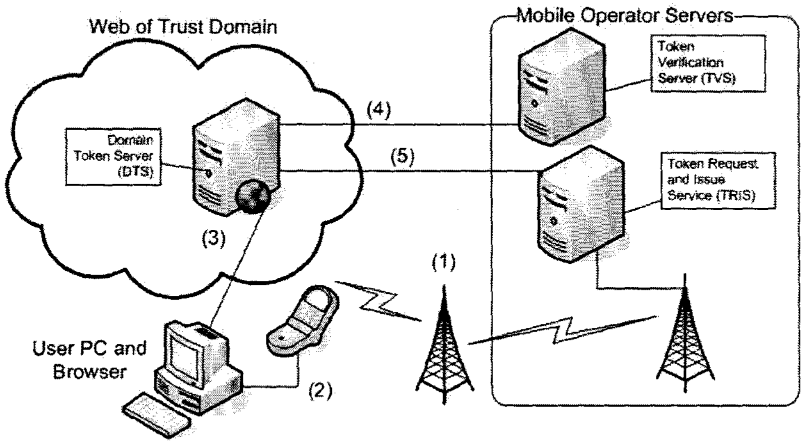
Web of Trust Domain
Domain Token Server (DTS)
Mobile Operator Servers
Token Verification Server (TVS)
Token Request and Issue Service (TRIS)
(1)
(2)
(3)
(4)
(5)
User PC and Browser

Figure 5.2: Overview of the framework

5.7.1 Mobile Phone and Mobile Operator Web Service

The architecture and security concerns of GSM networks, including General Packet Radio Service (GPRS), are well documented [63, 64, 89]. Access to the GSM network is controlled by the SIM card and its PIN code, where authentication of the user on the network generates a session key to prevent abuse, with encryption of communication over the radio interface, concealing the identities through temporary identity codes. Security methods in GSM protocols are standardised, specifying how each component interacts with each other and the range of their actions [63]. Standards allow for wide spread adoption and integration, giving GSM networks global relevance and applicability [37]. GPRS is a data value-added service that allows digital information to be sent and received across a mobile telephone network [62]. GPRS, using the 3G (third-generation) protocols, is suitable for transferring data over the GSM networks, and can support mobile web browsing and Internet applications [110].

In this framework (1) of Figure 5.2, an interface between the mobile operator web server and the mobile phone is created using GPRS to transmit data in the form of packets.

5.7.2 Mobile Phone and PC

One of the goals of a generic framework is that it should not rely heavily on the underlying implementation, but should rather integrate seamlessly regardless of the underlying architecture. In this framework (2) of Figure 5.2, it is essential to establish a connection between the Mobile Phone and the PC so that the potential of the Mobile Phone as a physical token can be realised. This can be achieved by either a direct connection over USB, Infra-Red or via a Windows technology such as Bluetooth. However, the generic nature of this approach will be diminished should an implementation of the framework be required to develop several versions of an application to facilitate the connection for each of the different operating system platforms and technology.

In order for a browser to interface with a mobile phone, the browser will have to make use of operating system specific services. Some browsers, such as Mozilla's FireFox, allow third parties to create plug-ins and extensions that can make use of operating system-based services without being specifically developed for the host operating system [115]. The approach employed in the FireFox browser is called XPCOM (Cross Platform Component Model) and is used to separate the implementation of a component from the interface, where an interface is a formalised communication channel. This is similar to Microsoft COM interfaces [30]. With such mechanisms, the browser becomes flexible client software that allows the framework to achieve relative freedom from operating system specific concerns.

5.7.3 User and Web of Trust Domain

In terms of the framework (3) of Figure 5.2, interactions between the user (via the web browser) and the WoT domain are either user-related or related to second factor authentication. User-related interactions are specific to the WoT domain, where the communication protocols used are of little concern. This section is to describe the interactions related to second factor authentication without becoming protocol and implementation specific.

Since the WoT domain may exist already, the second factor authentication functionality must be incorporated into the domain. A web service can be employed and integrated into the WoT domain such that it will provide an interface for the user to transport the GSM Token. Since usability is a goal, the web service is non-interactive, providing a level of autonomy where the user is only notified to the success or failure of the procedure together with suitable feedback. The browser plug-in should be enabled with an interface to interact with the web service on the WoT domain side.

The token is temporarily stored in the browser's memory space once the browser has acquired the GSM Token from the mobile phone. It is then important to transmit the token to the WoT domain so that the authentication process can continue. A GSM Token is created for a particular session with a particular WoT domain, where the browser will open a connection to that WoT domain's web service, based on the WebServiceHandle field. Once a suitably secure connection has been created, regardless of the underlying technology used, conceptually the token is passed to the DTS for it to be verified.

5.7.4 Web of Trust Domain and Mobile Operator Web Service

The connection between the WoT domain and the mobile operator Web Service is conducted in secret using a private back-channel over a public or private network. It is possible to create a secure point-to-point connection using SSL/TLS or IPSec technologies [38], providing features such as mutual authentication and confidentiality. The purpose of the back-channel connection is two-fold; one is to initiate a session request should a user try access protected resources, and the other is to verify the validity of a token. Should a user try access protected resources without first presenting a token, the DTS (5) of Figure 5.2 engages the "push" feature of the service by connecting to the TRIS on the mobile operator side. After mutual authentication, the DTS informs the TRIS of the request. The TRIS then builds up a request as described in Section 5.5.1.

Since the GSM Token arrives along the same path that is used by the user to connect to the WoT domain, it is necessary for the token authenticity to be verified. The WoT domain verifies the validity of any token received from a user through the use of the DTS. The DTS encrypts the received token with its own key such that the DTS can be authenticated by the TVS of the mobile operator. In fact, mutual authentication can occur when the secure point-to-point connection is created. The encryption of the token adds an additional layer of security. Based on the response of the TVS (4) of Figure 5.2, whether a token is valid or not, the requested access will be granted or denied to the user. The fact that a back channel is used to provide verification, where the token has value only after being verified, means that multi-path two factor authentication is achieved over two different network mediums.

5.8 Chapter Summary

The framework presented in this chapter supports the notion that the mobile phone is an ideal means to provide a pervasive and easily manageable two-factor authentication scheme over the Internet. As seen, the generic approach of this model allows greater applicability over many different domains, allowing for interoperability in terms of the De-Perimeterisation concept. Since increasing factors of authentication, and by creating a mechanism where the validation of the factors occur over different paths (networks), it increases the security of such an authentication process. An analysis and discussion of chapters four and five is presented in the next and final chapter before the conclusion.

Chapter 6

Analysis and Discussion

6.1 Chapter Overview

Chapter 4 presents a model for anonymity and a model for privacy. These models are in response to the issues discussed in Sections 3.3 and 3.2 respectively. Chapter 5 describes a framework incorporating a mobile phone as a second factor of authentication. This is based on the discussion in Section 3.4. In this section we analyse what each model does and discuss the assumptions and problems faced by each model in addressing their particular issue.

Section 6.2 analyses the model for anonymity. The model for privacy is discussed in Section 6.3. The mobile phone two-factor authentication is analysed and discussed in Section 6.4.

Before concluding this thesis, we will discuss a series of scenarios in Section 6.5. The goal is to show how these different models may be used together in some common scenarios. The chapter summary is available in Section 6.6.

6.2 Model for Anonymity

The issue of anonymity within an open environment is described in Section 3.3. We find that true anonymity is difficult to achieve since entities leave traces of their actions within a networked environment. Anonymity is a desirable feature because it protects the identities of parties in communication. A model is presented in Section 4.2 that depicts how anonymity can be achieved in an open environment. This section analyses how the goals of the model are achieved. The assumptions and problems faced by the model are also discussed.

What the Model Does

The model provides anonymity in both the message and the channel. Since each message is encrypted no identity information can be discerned from the message contents. However, the routing information concerning the addresses of the sender and receiver are still available. This is dealt within the channel by decoupling the direct connections between the sender and recipient through the use of an Identity Agnostic layer. A series of intermediate nodes are used to route messages between the sender and ultimate receiver. Such that, within the layer the sender from two nodes back is not known, as well as the destination after the next node is not known. The routing of messages within the layer is achieved with a degree of randomness so that the same path is not used twice in quick succession.

Accountability is achieved by extensive logging, thereby associating actions and transactions in the layer with the users and nodes that performed them. The information contained in the logs should be confidential (i.e. encrypted), in that only the system should be able to retrieve any meaningful data. It should be possible to perform an analysis of the log data, looking for anomalous behaviour of nodes and routing effectiveness.

Through using secure point-to-point connections, mutual authentication between nodes and clients occurs. Mutual authentication, where two entities in communication authenticate each other, is considered to be the better approach, as discussed in Section 2.3.3 and 3.2.2. Furthermore, encryption occurs with these types of connections [38].

Assumptions

The IA layer can be considered a Trusted Third Part (TTP), in that it is an entity that operates beyond the control of its participants and is based on a trust relationship stipulated by usage policies. The assumption is that the TTP will behave in its intended manner, that it will not observe the traffic and monitor user actions beyond the normal logging activities, and that it is not subvertable. A further fear is that the TTP will be in a position to fully observe the patterns of user behaviour, and sell this valuable information to the highest bidder. This may occur after a period of operation where ownership is transferred through sale or litigation. It is assumed that this will not occur, and that the IA layer is not under the control of essentially malicious parties.

Problems

Should the system be compromised, attackers would be unable to modify the logs due to encryption. However it might be possible to corrupt or delete the logs. This problem gives favour to making the logs centralised and hidden, as discussed in Section 4.2.1, although it might be possible to form an attack model when a node is compromised. This depends heavily on the protocol employed by the layer, as well as the manner in which individual nodes and machines are secured. Securing machines naturally entail the defence in depth approach, see Section 1.3.3. The nodes themselves provide points of attack, where malicious observers may routinely test the security capabilities of the nodes, looking for pliable weaknesses.

Functions that generate random numbers are not truly random [102], meaning that over long enough periods of observation, patterns may be discernable. This holds true in the case for any routing algorithm that employs pseudo-randomness if it is assumed that the environment is fully observable. Information can be gained through observing patterns and then using that to plan an attack on the system [67]. A possible solution to this problem, in the case of inter-nodal communication, is to broadcast to every node. Such that only one packet is valid while the rest of the packets are decoys. Since all the packets are encrypted the data will become stale before the attacker can find the correct, which involves breaking the encryption protocol and verifying the correct packet payload. However, the drawback with this approach is a severe increase in traffic overhead, becoming less scalable as more nodes are added.

6.3 Model for Privacy

This section analyses what the model for privacy does, including the assumptions and the problems faced. The issue of privacy is described in Section 3.2. Privacy is seen to be the control over the disclosure and usage of personal information, specifically the protection of that information not being used outside the interests of the owner of that information. A model for privacy is presented in Section 4.3 that attempts unify control of credentials from disparate accounts in a *user centric* approach.

What the Model Does

Privacy is attained by following the *user centric* paradigm of meta-identity, which is discussed in Section 2.2.4. This entails giving discretional control over personal information to the user instead of the administration. *User centric* credential management means that users can protect their personal information by deciding what credentials to disclose and when to disclose them. Fine-grained control is given by unifying the management of disparate accounts from different domains into a single system. Single Sign-On is side benefit meta-identity systems where users authenticate to the system itself, which acts as a proxy for the user's other accounts. Mutual authentication is once again achieved by employing secure point-to-point technologies.

Furthermore, the protection of one's privacy can be achieved by trying to limit the amount of credentials disclosed. By creating accounts where credentials are cryptographic derivations based on a master account, the specific credential is not disclosed. This follows the same principle of pseudonyms, as discussed in Section 3.2.1, where the link between the pseudonym and the real identity is kept hidden or secret. In the cases where a domain allows purely derived accounts anonymity is achieved, since the domain bases the account on the trust relationship with the portal. And since the portal acts as a proxy the derived account is decoupled from the identity of the user.

Assumptions

The operation of such a model depends on the trust relationships built around the meta-identity portal. It is assumed that the Identity Providers and Service Providers are willing to enter into a trust relationship with the portal. In that, the IdP and SP are willing to accept log-in information from the portal. This hinges on the ability of the portal to provide the connection between the derived account and a bona fide identity should the need arise.

It is assumed that for some domains the use of derived accounts is acceptable. In that the IdP will allow the portal to store personal information locally instead of with the IdP. The IdP will receive cryptographic derivatives of the types of credentials the IdP requires in order for an account to be created.

It is further assumed that since the portal acts as a TTP for all parties involved the portal is able to act as a certifying authority. For some types of accounts this might extend to the real world certification of an individuals credentials.

Problems

A problem for central certification schemes is "who certifies the root", a recursive problem where at some point a level of trust is going to be invested into an entity. This is discussed in Section

2.2.5. In order to create an environment where trust in a certification authority is acceptable a suitable infrastructure needs to be in place. In that, the portal may need to be associated with an entity that is inherently trustworthy, specifically an entity that is not inherently self-serving. That means that the certifier of this identity is required to be an authority that has real and fixed ties to the real world and is in a position to enforce that authority. In that, credible proof presented in person is included in the creation process of the master account which is certified by the authority. For example, telecommunication companies, including both fixed-line and mobile operators, and financial institutions. Both have considerable ties to real world assets, as well as an invested concern in the economy in which they reside. So the problem here is the lack of an infrastructure and controls in which the meta-identity portal scheme can flourish.

By moving personal information and incorporating disparate IdP account control to a single location, the portal becomes a single point of attack and failure. It may become necessary to incorporate multiple factors of authentication into the log-in process. To avoid theft and inadvertent disclosure the stored information should be necessarily encrypted with the cryptographic keys stored elsewhere. To mitigate attempted destruction or corruption of the information, a backup plan should be incorporated into the portal scheme. Furthermore, a distributed approach, where different types of information, credentials are stored in different location with different process to retrieve them, may introduce greater complexity. So instead of storing the complete set of credentials belonging to a user in a single location, several locations would be used to store different subsets of the credentials.

A problem arises when the Portal architecture becomes a single point of failure. If the Portal system (whether it be a single machine or a distributed cluster) were to be compromised and the account data contained copied or deleted, the system as a whole would be useless or defunct. This can be mitigated by ensuring proper measures are taken to achieve data separation and segregation, and by removing the direct link between various sets of credentials and accounts belonging to the same user. The Portal system may also be susceptible to Denial of Service attacks. By interrupting normal service to users, which may render linked accounts unusable, it would reduce confidence in the system. This is a universal issue, applicable to every system in a networked environment.

6.4 Mobile Phone Two Factor Authentication Framework

In order to provide a pervasive multi-factor authentication scheme for an environment such as the Internet, one has to consider approaches in light of the user experience. A desirable scheme will be an approach that incorporates increased security without sacrificing usability. Section 3.4 covers the different types of factors that can be used in the authentication process. The mobile phone is shown to be an ideal candidate as a security token, which forms the basis for the framework presented in Chapter 5. The focus is to provide a generic framework that has general applicability within the Internet, as opposed to being restricted to a single domain or system. This section analyses what the framework does, and describes the assumptions and problems faced.

What the Framework Does

The framework provides a generic and pervasive two-factor authentication scheme for use in an open environment. An aim of the framework is to extend the general applicability of such an approach, making it suitable for a wide range of systems and domains.

As a second factor of authentication, the first being the standard user-name and password pair, the mobile phone is well suited for a number of reasons. Though discussed in Section 3.4.3, the mobile phone is a pervasive device that has its own computational and memory capabilities. The mobile phone is solidly entrenched in the lifestyle of the user, and is typically kept on their person at all times. One just needs to consider one's own habits concerning the mobile phone. As other physical tokens go, the mobile phone does not suffer from the same token management drawbacks of wide-scale deployment schemes. All the user requires is a specific SIM card that is issued by the mobile operator. The SIM controls access to the GSM network, as it contains the authentication details of the subscriber. The SIM will contain the application that communicates with mobile operator Token Request and Issue Service (TRIS), which is responsible for tracking session requests and issued tokens that describe authorised sessions. An application on the Mobile Equipment side provides the user interface for the SIM application. This is because the SIM is a self-enclosed security domain, where access is through a published specification, in this case the SIM Application Toolkit (SAT).

The incorporation of the GSM network is an additional network that is used in the authentication process. Firstly when the user authenticates to a domain via the Internet, entailing a single path, a cryptographic token is used to describe the session. Secondly the user initiates a second phase using the mobile phone, which returns a second cryptographic token. The token is transmitted to the domain via the PC and Internet. By having a private back-channel connection between the domain and the mobile operator Token Verification Service (TVS), where the second token is verified, the second path is completed. In this framework, two paths are used to complete authentication process, thereby assuring two-factor authentication over multiple paths.

The framework allows for both a credential push and pull, meaning that a session may be initiated by the user or the domain. A user simply engaging in activities within the domain may warrant the domain contacting the TRIS, which fires off an Alert to the subscriber's mobile phone. This has the feature of alerting the user to unauthorised access of the domain should the user's account be compromised. The other way entails a user initiating a session from the mobile phone.

Assumptions

The security of the framework depends on the assumption that the SIM and GSM security features are strong enough to mitigate the risk that a compromise of either will result in a complete compromise of the system. In that, the security of the SIM or the GSM network are sufficiently decoupled that should either be compromised it will not compromise the entire system.

For this framework to be feasible it is assumed that the mobile operators are willing to engage in such a service. Additionally, it is assumed that both the mobile operator and the various domains are inclined to enter into a trust relationship with each other. Ideally, the mobile operators offer the service to both subscribers and domains.

Problems

Although the SAT is based on a specification [1], there is an issue concerning the interoperability of the implementation from different vendors. This may result in a SIM vendor-specific implementation of the SIM application. Different mobile operators, who may not necessarily use the same SIM vendor, may use different implementations of the SIM application.

Though the following are not problems *per se,* they do represent security concerns for the framework. For any system, there will be certain points of failure that will prove critical to the system's success. Examining these points and providing contingency plans is vital. The detection of, and recovery from, the compromise of components not directly in control of the user should be the responsibility of the system. The environment of the user is restricted to the standard "Best Practices" controls, such as current anti-virus software and operating system hardening techniques.

Loss of phone

Due to the habits of mobile phone subscribers, people are prone to notice the loss of a phone quickly. Furthermore, the typical behaviour of a person with a lost phone is to contact the police and the mobile operator to report the loss (or theft), and have the phone blacklisted. Blacklisted phones, using the IMEI number, are unable to connect to the mobile network. Should the phone be lost or stolen, upon the notification of the loss, the user's account will be suspended until a new phone and SIM card is issued. Mobile phone theft is a common petty crime around the world, with mobile companies releasing phone software that detects whether or not a phone is stolen, which can delete the phone's data.

Compromise of User machine and Phone

Any computing platform is susceptible to different forms of attack, from the network to more sophisticated software, of which it is up to the user to follow the "Defence in Depth Best Practices" in terms of securing a machine and operating system. Running updated anti-virus and spyware software is one necessary method. Should there be any evidence of a machine or a phone being compromised, usage should cease immediately, and correction procedures followed.

6.5 Scenario Discussion

This section attempts to bring the models presented in Chapters 4 and 5 together in a series of scenarios. This is the essence of de-perimeterisation, as discussed in Section 1.3.2. De-perimeterisation is a security design paradigm for open environments centred on secure communication and unhindered information flows. As guiding principles, it is a set of solutions that are modular and decoupled components that focus on interoperability based on open standards.

The three scenarios that are discussed increase in security requirements and risk. The purpose is to demonstrate the benefits of the models in the face of adversity in a hypothetical manner.

6.5.1 Setting the Scene

Bob is an average user of the Internet, in that he engages in general web browsing, and from time to time makes on-line purchases of goods and services. Bob is also a mobile subscriber with

Alpha Mobile. Bob decides to make use of the two-factor authentication service offered by his mobile operator. Bob is issued with an updated SIM card for his mobile phone, and an account linked to his phone. The cell phone application that allows access to the phone's security token functionality is accessible through the use of a PIN.

Bob decides to link his on-line banking account to the services offered by Alpha Mobile. Due to the nature of the relationship between Alpha Mobile, the bank allows Bob to authorise transactions through his mobile phone. The bank can issue challenges to Bob via the Alpha Mobile service and application residing on his phone. In order to proceed with the transaction Bob will have to respond with the appropriate information. Bob is alerted to any bank transactions pending his authorisation, of which he can decide to accept or decline.

As it so happens, Bob opens up an account with the meta-identity Portal service administered by Primary Privacy. Bob decides to let his mobile phone subscriber account become the master account, allowing his phone to become a physical security token with that account. To achieve this Bob visits the nearest retail office of his mobile operator and presenting physical proof of his legal existence. The mobile operator issues a physical certificate containing his credentials, which is created from the lowest level of abstraction of an individuals real world identity. Bob then goes to the offices of Primary Privacy with mobile phone, certificate and legal documents in hand to initiate the creation of the Portal master account. The master account is used to administer and access Bob's other accounts, and is accessed through a user-name/password pair.

In addition to the security measures taken above, Bob opts to make use of an Identity Agnostic service offered through Capital, a TTP. Capital anonymises the communication traffic between Bob and the client nodes that interact directly with Alpha Mobile and the web sites that Bob surfs. It is given that Bob implicitly connects directly to a participating Capital Secure Anonymous Server (SAS) in order to access the Internet for both general web browsing and services.

The Perspective of the Attacker

Within the environment lurks Simon, a malicious individual, who is intent on building up a comprehensive identity profile on Bob and his activities. For the sake of discussion, Simon does not know any particulars about Bob, nor with whom he banks, nor his current mobile operator. If this information were known to Simon it would be a case of preparing to attack the points of contact within the environment such as Capital, Primary Privacy and Alpha Mobile servers.

Simon has competent skills in active and passive based methods of information acquisition. Simon is also well placed to intercept traffic between Bob and his initial point of contact with the

Internet. Simon, however, may or may not be able to compromise the machines known as points of contact, as discussed above.

Simon is initially aware of the traffic originating from Bob's connection to the Internet, this is the first point of contact. Due to Bob using the IA service from Capital, the traffic is necessarily encrypted. Simon tracks the destination of the outbound traffic, as well as the origination of the inbound traffic. Simon cannot discern the nature of the communication at first. After some investigation Simon uncovers the interaction between Bob and the SAS belonging to Capital. This can be achieved by correlating the IP addresses of the outbound traffic to known Internet services. This becomes the second point of contact. Without compromising and reverse-engineering the IA system protocol Simon will not be able to discover the ultimate destination of Bob's traffic.

Low-Level Security: The Community Forums

Bob joins an on-line community for developers, requiring registration to control access to otherwise free resources and public spaces. Bob is required to disclose his name, email address and other non-critical information. The community web site has an established trust relationship with Primary Privacy, the Portal service. As such, the Portal account is deemed sufficient for registration by allowing the details derived from the master account to be used. In that Bob chooses a username, acting as a pseudonym, to link to the master account. The details that are required by the community web site are then cryptographically derived from the master account. Bob assigns this account to a low security level, only requiring Bob to log on to his Portal account. At this level, the Portal provides Single Sign-On functionality, as well as protecting his privacy.

If Simon were to discover the community site as a possible third point of contact, preparations can be made to actively pursue Bob's relationship with the site. At this juncture, Simon acquires the entire user base of the community site. This does not help as Bob's account is protected by a pseudonym which is linked to his Primary Privacy Portal account. Since the details of Bob's account are obfuscated through encryption, there are no apparent correlating links that Simon can find. If Simon were to go through the laborious process of tracking each individual's activity then Bob's identity would still be protected by the IA service.

In the absence of the IA service Bob's interaction would be revealed through linking Bob's account and the first point of contact. Having access to the user data would allow Simon to make the connection between Bob's community account with Primary Privacy's Portal service. This would allow Simon to add the fourth point of contact to Bob's identity profile.

Medium-Level Security: The Amazon.com Connection

Primary Privacy has an established trust relationship with Amazon.com, where Bob shops regularly. Bob decides to link his Amazon account to his Portal account, which is deemed to be of a middle security level. Bob logs into the Portal, indicating a desire to access his Amazon account. The Portal acts as a proxy, opening up a connection to Amazon.

When Bob is ready to make a purchase, instead of submitting a valid credit card number, he indicates payment via his bank account. The bank then sends a challenge to Bob on his phone requesting authentication for his intention to make a payment for that amount. Amazon.com will only ship orders once confirmation of payment has been received.

In this scenario Simon still has two points of contact; Bob's Internet connection and the network of Capital Secure Anonymous Servers. The other possible points of contact are the Amazon servers, Bob's phone, the bank and the Portal server.

If Simon could compromise Bob's Amazon account then Simon would be able to glean the contact information Bob had disclosed to Amazon, such as his postal and email address, and mobile number. Simon could then conduct a search of the area to find what banks operated there. Simon could employ phishing attacks against Bob to have him disclose banking details, or tempt Bob into installing malicious applications on his mobile phone. However, if Bob had opted to assign an appropriate security level to his Amazon account, then use of that account could be monitored and protected by the Portal service. This may reduce the risk of identity theft by incorporating the services of Alpha Mobile.

High-Level Security: The Bank

Bob does most of his bank transactions on-line. His bank account is linked to his portal Master Account, which he has marked as a high security level. In order to ensure the additional security, Bob can only access his on-line bank account using the combination of cell phone and master account. Bob logs on to the Portal to access the master account, indicating a desire to access his bank account. At the same time, Bob uses his cell phone to request a session for his bank account. Only when these corresponding two requests are made can Bob access his bank account to engage in transactions.

Simon is aware that to compromise both the connection between Bob and his bank, and the connection between Bob's Alpha Mobile two factor authentication service and the bank would take exponentially more work than compromising either. As such, fraud is easier to commit through identity theft rather than outright cracking systems. If, at this point, Simon had managed to glean Bob's banking details and attempted to defraud Bob of his money by transferring to another account, or making payments using Bob's account, then Simon would fail. Bob would be alerted to this criminal activity through his mobile phone when it requests payment or transfer authorisation.

6.6 Chapter Summary

For each of the models presented in this thesis we have analysed and discussed what the model does, as well as assumptions made and the problems faced. The model for anonymity is shown to depend on the TTP that comprises the Identity Agnostic layer. Transaction logging and the providing of accountability depends on the security of the layer, both in terms of the protocols and machines performing the logging. The model can provide a level of anonymity, protecting the contents of the message as well as the identity of participants within the communication channel.

The model for privacy depends on the ability to create trust relationships between the domains and a centralised or distributed meta-identity portal. In that whether the use of derived accounts are acceptable or not depends on the nature of an account within a domain. The master account concept suffers from the same issue as certifying authorities, the issue regarding that which forms the root of a trust relationship. However, the *user centric* paradigm associated with meta-identity still provides a level of privacy for the user.

The mobile phone two-factor authentication framework depends on the strength of the SIM and GSM security features and as well as the willingness of mobile operators to engage in such a scheme. Besides the interoperability issues facing the implementations of the vendor-specific SAT, the framework provides a scheme using a pervasive device firmly entrenched in the user lifestyle. The approach itself can be used to increase the security of any domain within an open environment.

The scenario discussion brought the models together in an attempt to illustrate possible application. As such the models can be applied in a manner of different ways so as to operate independently of one another without impeding one another. However, the real value comes from the synergy of allowing the models to complement each other. As each model has weaknesses or short-comings, it may be possible to co-ordinate the models in such a way to overcome them.

The conclusion of the thesis is presented in the final chapter.

Chapter 7

Conclusion

7.1 Problem Statement Revisited

The problem statement, presented in Section 1.2, gives the goals and aims of the thesis. Specifically, this thesis performs an investigation into issues concerning Identity and Access Management solutions, with a focus on open environments. These reduce to anonymity, privacy and multi-factor authentication within an open environment.

Section 3.3 describes the concept of anonymity, as it is a concern for users and systems within a networked environment, in that the security measures can be thwarted through the observation of the environment. By an attacker observing the communication channel and monitoring the interactions between users and systems over a long enough period of time, it is possible to infer knowledge about the users and systems. This information can be used to compromise user and system security. Approaches to providing anonymity are explored.

Privacy entails the user account representation and management, such that the disclosure and usage is a function of control of the owner of the information. Section 3.2 addresses the idea of privacy and how it can be achieved in an open environment. Once information is published or divulged on the network, there is very little way of controlling the subsequent usage of that information. Privacy is identified to be important in two areas; where and how personal information is stored, and how it is disclosed.

In Section 3.4 the inherent weakness of single factor authentication mechanisms is addressed. By increasing the factors used in authentication, the amount of work required to compromise the system increases non-linearly. Within an open network, several aspects hinder wide scale adoption and use of multi-factor authentication schemes, such as token management and the impact on usability. Biometric approaches, due the costs involved as well as perceived invasiveness, are shown to be unsuitability for wide scale deployment. There is a discussion surrounding the mobile phone as security token, supporting the notion that it best suits the role.

The identification of the above mentioned anonymity and privacy issues result in the development of models that attempt to address these issues. A generic framework, derived using open systems, is used to solve the pervasive multi-factor authentication problem. The models and framework are analysed and discussed.

7.2 Achievements

As per the issues declared in the problem statement, the thesis focuses on addressing these issues from a high level of abstraction with reference to real world implementations. Chapter 4 presents the models that address anonymity and privacy.

Providing anonymity, presented in Section 4.2, is a matter of protecting the contents of the message though encryption, and hiding the identities of the sender and receiver of the message. This is achieved by creating an Identity Agnostic layer that effectively obfuscates the communication environment through decoupling the direct connection between the sender and receiver. Furthermore, by incorporating decoy traffic and introducing complexity to the routing mechanisms of the layer, the problem of prolonged observation is mitigated. Accountability, the converse of anonymity, is provided by secure transaction logging within the layer.

Meta-identity is shown to embody the *user centric* paradigm, with the development of a model that incorporates the unification of control of disparate accounts into a single location. The model is described in Section 4.3, where privacy is achieved by giving control over the use and disclosure of personal information to the user. By developing the concept of master and derived accounts, it is possible to provide further privacy through the non-disclosure of credentials. However, this depends on the ability to create trust relationships between the domains and a centralised or distributed meta-identity portal.

The mobile phone two-factor authentication framework is presented in Section 5. The mobile phone is shown, in co-operation with the mobile operator and GSM network, to be an ideal security token. Having its own computational and memory capabilities, makes the mobile phone flexible, adaptable and extremely useful. As it is a pervasive device firmly entrenched in the user lifestyle, it does not impact on the user experience. The approach itself can be used to increase the security of any domain within an open environment.

Chapter 6 performs the analysis of the each of the models. It provides the assumptions and problems faced by each model. A discussion, based on a series of scenarios, shows how the models may operate together, following the paradigm of De-P. However, since these models are brought together in a series of hypothetical scenarios, it is difficult to accurately judge their applicability to real world situations from a technical perspective. Though, as models, there is value in adopting a multi-faceted approach to security, privacy and anonymity. Approaches to resolving these issues should be layered yet be independent, and interoperable.

The goal of this thesis is to identify key issues in the open environment, and as a result the models presented in Chapters 4 and 5 are developed. A technical implementation would be subject to the limitations of the technology used in that implementation. As such, a high level of abstraction is used when dealing with the models.

7.3 Future Work

There is much scope for future work based on this thesis. Firstly, there is scope in implementing a framework of the model for anonymity. This can entail creating the Identity Agnostic layer in which the different implementations mentioned in Section 3.3.1 can be used. The measuring of bandwidth usage and the communication overhead, the monitoring of the routing decision algorithm and other information can rate the fitness of the different approaches. There is also scope in implementing a fresh approach, based on the Agnostic Identity layer.

Secondly, future work can done by exploring the concept of meta-identity systems especially within the context of open environments such as the Internet. There is room for exploring the conditions required for the general acceptance of the master/derived account concept. This entails canvasing the different types of Identity Providers to gauge what their requirements are in order to achieve such an approach. Then indeed there is scope for implementing a meta-identity system and evaluating it against available systems.

Thirdly and finally, there is scope in implementing the generic mobile phone two-factor authentication framework. However, this requires the backing and support of a mobile operator for access to the needed equipment and technology components.

References

[1] 3GPP. Specification of the sim application toolkit for the subscriber identity module - mobile equipment (sim - me) interface (gsm 11.14). Technical Specification Version 8.17, 3rd Generation Partnership Project, 1999.

[2] Rania Abdelhameed, Sabira Khatun, Borhanuddin Mohd Ali, and Abdul Rahman Ramli. Authentication model based bluetooth-enabled mobile phone. *Journal of Computer Science*, 1:200–203, 2005.

[3] Karl Aberer, Anwitaman Datta, and Manfred Hauswirth. A decentralised public key infrastructure for customer-to-customer e-commerce. *Int. J. Business Process Integration and Management*, 1(1):26–34, November 2005.

[4] Gail-Joon Ahn and John Lam. Managing privacy preferences for federated identity management. In *DIM '05: Proceedings of the 2005 workshop on Digital identity management*, pages 28–36, New York, NY, USA, 2005. ACM Press.

[5] Couto V. Disher C Alvarez, E. Business process outsourcing & offshoring. *EuroMoney*, August 2003.

[6] Holte M. Andrews, W. How web services provide roi. Report, May 2003.

[7] Michael Backes and Thomas Gross. Tailoring the dolev-yao abstraction to web services realities. In *Proceedings of the 2005 workshop on Secure web services*, pages 65–74, New York, NY, USA, 2005. ACM Press.

[8] P. Bahl, S. Venkatachary, and A. Balachandran. Secure wireless internet access in public places. In *Communications, 2001. ICC 2001. IEEE International Conference on*, volume 10, pages 3271–3275, 2001.

[9] Vicente Benjumea, Javier Lopez, Jose A. Montenegro, and Jose M. Troya. A first approach to provide anonymity in attribute certificates. *Public Key Cryptography*, 2974:402–415, 2004.

[10] Elisa Bertino, Elena Ferrari, and Anna Squicciarini. Trust negotiations: Concepts, systems, and languages. *Computing in Science and Engg.*, 6(4):27–34, 2004.

[11] Elisa Bertino, Elena Ferrari, and Anna Cinzia Squicciarini. Trust-X: A peer-to-peer framework for trust establishment. *IEEE Transactions on Knowledge and Data Engineering*, 16(7):827–842, 2004.

[12] Abhilasha Bhargav-Spantzel, Jan Camenisch, Thomas Gross, and Dieter Sommer. User centricity: a taxonomy and open issues. In *DIM '06: Proceedings of the second ACM workshop on Digital identity management*, pages 1–10, New York, NY, USA, 2006. ACM Press.

[13] Abhilasha Bhargav-Spantzel, Anna Squicciarini, and Elisa Bertino. Integrating federated digital identity management and trust negotiation. In *review IEEE Security and Privacy Magazine*, 2005. CERIAS TR 2005-46.

[14] Abhilasha Bhargav-Spantzel, Anna C. Squicciarini, and Elisa Bertino. Establishing and protecting digital identity in federation systems. In *DIM '05: Proceedings of the 2005 workshop on Digital identity management*, pages 11–19, New York, NY, USA, 2005. ACM Press.

[15] Mohit Bhargava and Catuscia Palamidessi. Probabilistic anonymity. *Lecture Notes in Computer Science*, 3653:171 – 185, 2005.

[16] Matt Blaze, Joan Feigenbaum, and Jack Lacy. Decentralized trust management. In *Proceedings of the 1996 IEEE Symposium on Security and Privacy*, page 164, Washington, DC, USA, 1996. IEEE Computer Society. SP '96: Proceedings of the 1996 IEEE Symposium on Security and Privacy.

[17] N Bleach. Visioning white paper. Jericho Forum White Paper., 2005.

[18] Katrin Borcea-Pfitzmann, Marit Hansen, Katja Liesebach, Andreas Pfitzmann, and Sandra Steinbrecher. What user-controlled identity management should learn from communities. *Information Security Technical Report*, 11(3):119–128, 2006.

[19] Kim Cameron. The laws of identity. Microsoft White Paper http://msdn.microsoft.com/webservices/understanding/advancedwebservices/default.aspx?pull=/library/en us/dnwebsrv/html/lawsofidentity.asp., May 2005.

[20] Kim Cameron and Michael B. Jones. Design rationale behind the identity metasystem architecture. Microsoft Corporation White Paper, February 2006.

[21] Paul Campbell, Ben Calvert, and Steven Boswell. *Security+ Guide To Network Security Fundamentals*. Thomson, 2003. ISBN: 0-619-21294-2.

[22] Scott Cantor, Jeff Hodges, John Kemp, and Peter Thompson. Liberty id-ff architecture overview version 1.2. Technical report, Liberty Alliance Project, 2003.

[23] Scott Cantor, John Kemp, Rob Philpott, and Eve Maler. Assertions and protocols for the oasis security assertion markup language (saml) v2.0. Security services technical committee, OASIS Standards, 2005.

[24] David W. Chadwick and Alexander Otenko. The PERMIS X.509 role based privilege management infrastructure. *Future Generation Computer Systems*, 19(2):277–289, February 2003.

[25] Chin-Chen Chang and Jung-San Lee. An anonymous voting mechanism based on the key exchange protocol. *Computers & Security*, 25(4):307 – 314, 06 2006.

[26] W.R. Cheswick and S.M. Bellovin. *Firewalls and Internet Security: Repelling the Wily Hacker*. Addison-Wesley Publishing Company, Reading, Massachusetts, 1st edition, 1994.

[27] Dwaine Clarke, Jean-Emile Elien, Carl Ellison, Matt Fredette, Alexander Morcos, and Ronald L. Rivest. Certificate chain discovery in spki/sdsi. *J. Comput. Secur.*, 9(4):285–322, 2001.

[28] Ian Clarke, Oskar Sandberg, Brandon Wiley, and Theodore W. Hong. Freenet: A distributed anonymous information storage and retrieval system. *Lecture Notes in Computer Science*, 2009:46–52, 2001.

[29] Sebastian Clauss and Marit Kohntopp. Identity management and its support of multilateral security. *Computer Networks*, 37(2):205–219, October 2001.

[30] Microsoft Coporation. http://microsoft.com, accessed may 2007. Available Online, 2007.

[31] Mark Crosbie. Biometrics for enterprise security. *Network Security*, 2005(11):4–8, November 2005.

[32] Anwitaman Datta, Manfred Hauswirth, and Karl Aberer. Updates in highly unreliable, replicated peer-to-peer systems. In *In the proceedings of the 23rd International Conference on Distributed Computing Systems, ICDCS2003 (to appear)*, 2003.

[33] Barbara I. Dewey, Peter B. DeBlois, and the EDUCAUSE Current Issues Committee. Current it issues survey report 2006. *Educause Quarterly*, 29(2):12 – 30, 2006.

[34] I. Djordjevic and T. Dimitrakos. A note on the anatomy of federation. *BT Technology Journal*, 23(4):89–106, 2005.

[35] Kirsten Doyle. Fnb fights phishing. http://www.mydigitallife.co.za. Available Online, May 2007.

[36] C. Ellison, B. Frantz, B. Lampson, R. Rivest, B. Thomas, and T. Ylonen. Rfc 2693: Spki certificate theory. Technical report, SPKI Working Group, United States, 1999.

[37] Hannes Federrath. Protection in mobile communications. In *Multilateral Security in Communications*, pages 349–364. Addison-Wesley-Longman, 1999.

[38] Niels Ferguson and Bruce Schneier. *Practical Cryptography.* Wiley Publishing, Inc., 2003. isbn: 0-471-22894-x.

[39] B. J. Fogg and Hsiang Tseng. The elements of computer credibility. In *CHI '99: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 80–87, New York, NY, USA, 1999. ACM Press.

[40] P. Foote and T. Neudenberger. Beyond sarbanes-oxley compliance. *Computers & Security*, 24(7):516–518, October 2005.

[41] OASIS (Organization for the Advancement of Structured Information Standards). Oasis security services tc. http://www.oasis-open.org/committees/security/faq.php, accessed may 2007. Available Online, 2007.

[42] Internet Engineering Task Force. http://www.ietf.org, accessed january 2007. Available Online, 2007.

[43] Jericho Forum. http://www.opengroup.org/jericho. Available Online, 2007.

[44] Wireless World Forum. http://www.w2forum.com. Available Online, March 2007.

[45] Steven Furnell. Handheld hazards: The rise of malware on mobile devices. *Computer Fraud & Security*, 2005(5):4–8, May 2005.

[46] Diego Gambetta. *Can We Trust Trust?*, chapter Chapter 13, pages 213–237. Basil Blackwell, 1988. Reprinted in electronic edition from Department of Sociology, University of Oxford, chapter 13, pp. 213-237".

[47] O. Goldreich. Zero-knowledge twenty years after its invention, 2002.

[48] The Liberty Alliance Technology Expert Group. The development of open, federated specifications for network identity. *Information Security Technical Report*, 7(3):55–64, 2002.

[49] The Open Group. http://www.opengroup.org. Available Online, 2007.

[50] R. Guha, Ravi Kumar, Prabhakar Raghavan, and Andrew Tomkins. Propagation of trust and distrust. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 403–412, New York, NY, USA, 2004. ACM.

[51] The TCP/IP Guide. Standards. http://www.tcpipguide.com/free/t_proprietarydefactoandopenstandards.htm accessed march 2006. Available Online, September 2005.

[52] T Harbour. Defence - on the home front. Security reading room, SANS Institute, 2003.

[53] Amir Herzberg, Yosi Mass, Joris Michaeli, Yiftach Ravid, and Dalit Naor. Access control meets public key infrastructure, or: Assigning roles to strangers. In *SP '00: Proceedings of the 2000 IEEE Symposium on Security and Privacy*, page 2, Washington, DC, USA, 2000. IEEE Computer Society.

[54] T. Hiller, P. Walsh., X. Chen, M. Munson, and G. Dommety. Rfc 3141 - cdma2000 wireless data requirements for aaa. Technical report, The Internet Society, 2001.

[55] Jeff Hodges, Rob Philpott, and Eve Maler. Glossary for the oasis security assertion markup language (saml) v2.0. Security services technical committee, OASIS Standards, 2005.

[56] P. Holmes. An introduction to boundaryless information flow. Open Group White Paper, 2002.

[57] R. Housley, W. Polk, W. Ford, and D. Solo. Rfc 3280: Internet x.509 public key infrastructure certificate and certificate revocation list (crl) profile. Network working group, The Internet Society, april 2002.

[58] CT Howie, J Neethling, I Currie, C da Silva, C Duval, B Grant, A Grobler, M Heyink, S Jagwanth, and A Tilley. Privacy and data protection, discussion paper 109, project 124. Online: www.doj.gov.za/salrc/index.htm, October 2005.

[59] Internet2. Shibboleth project. http://shibboleth.internet2.edu/. Available Online, may 2007.

[60] Audun Josang, Roslan Ismail, and Colin Boyd. A survey of trust and reputation systems for online service provision. *Decision Support Systems*, 43(2):618–644, March 2007.

[61] Audun Josang and Simon Pope. User centric identity management. In *In Proceedings of AusCERT Conference*, 2005.

[62] Sherif Kamel and Khaled Wahba. Gprs security as a qos in the telecommunication industry case of vodafone egypt. *International Journal of Information Management*, 24(1):5–27, February 2004.

[63] Sumit Kasera and Nishit Narang. *3G Mobile Networks*. Tata McGraw-Hill Publishing Company, 2004.

[64] B. Kasim and L. Ertaul. Gsm security. In *Proceedings of the 2005 International Conference on Wireless Networks, ICWN'05*, Las Vegas, June 2005.

[65] Dogan Kedogan, Dakshi Agrawal, and Stefan Penz. Limits of anonymity in open environments. In *IH '02: Revised Papers from the 5th International Workshop on Information Hiding*, pages 53–69, London, UK, 2003. Springer-Verlag.

[66] Y. Keleta, M. Coetzee, JHP. Eloff, and HS. Venter. Proposing a secure XACML architecture ensuring privacy and trust. In *Proceedings of the 5TH Annual International Information Security South Africa (ISSA) conference*, July 2005. 2005 Keleta, Y, Coetzee, M., Eloff, JHP, Venter, HS, (2005), Proposing a Secure XACML architecture ensuring privacy and trust, Proceedings of the 5TH Annual International Information Security South Africa (ISSA) conference, July 2005, ISBN 1-86854-625X, Johannesburg, South Africa.

[67] D. Kesdogan and C. Palmer. Technical challenges of network anonymity. *Computer Communications*, 29(3):306–324, February 2005.

[68] Dorene L. Kewley and John Lowry. Observations on the effects of defense in depth on adversary behavior in cyber warfare. In *IEEE Workshop on Information Assurance and Security*, June 2001. Proceedings of the 2001 IEEE Workshop on Information Assurance and Security United States Military Academy, West Point, NY, 5-6 June, 2001.

[69] K Krechmer. Open standards requirements. In *HICSS: Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, pages 204b–204b, 2005. 03-06 Jan. 2005.

[70] Gregg Kreizman, Ant Allan, Roberta J. Witty, Ray Wagner, John Enck, Neil MacDonald, Barry Runyon, John Pescatore, Avivah Litan, Eric Ouellet, and Vic Wheatman. Hype cycle for identity and access management technologies, 2006. Technical report, Gartner, June 2006.

[71] Jeff Langenderfer and Stefan Linnhoff. The emergence of biometrics and its effect on consumers. *Journal of Consumer Affairs*, 39(2):314–338, 2005.

[72] P Langlois. From disappearing boundaries to security governance. http://www.continuitycentral.com/feature080.htm. Available Online, 2004.

[73] Adam Lee, Jodie Boyer, Lars Olson, and Carl Gunter. Defeasible security policy composition for web services. In *The 4th ACM Workshop on Formal Methods in Security Engineering (FMSE 2006)*, 2006.

[74] Adam J. Lee, Marianne Winslett, Jim Basney, and Von Welch. Traust: a trust negotiation-based authorization service for open systems. In *SACMAT '06: Proceedings of the eleventh ACM symposium on Access control models and technologies*, pages 39–48, New York, NY, USA, 2006. ACM Press.

[75] Raph Levien. http://www.advogato.org/trust-metric.html. Available Online, May 2006.

[76] M. Looi. Enhanced authentication services for internet systems using mobile networks. In *Global Telecommunications Conference, 2001. GLOBECOM '01. IEEE*, volume 6, pages 3468–3472, 2001.

[77] Javier Lopez, Rolf Oppliger, and Gunther Pernul. Authentication and authorization infrastructures (AAIs): a comparative survey. *Computers & Security*, 23(7):578–590, October 2004.

[78] Javier Lopez, Rolf Oppliger, and Gunther Pernul. Why have public key infrastructures failed so far? *Internet Research*, 15(5):544–556, October 2005.

[79] Markus Lorch, Seth Proctor, Rebekah Lepro, Dennis Kafura, and Sumit Shah. First experiences using XACML for access control in distributed systems. In *XMLSEC '03: Proceedings of the 2003 ACM workshop on XML security*, pages 25–37, New York, NY, USA, 2003. ACM Press.

[80] Mingchao Ma and Steve Woodhead. Authentication delegation for subscription-based remote network services. *Computers & Security*, 25(5):371–378, July 2006.

[81] John A. MacDonald and Chris J. Mitchell. Using the gsm/umts sim to secure web services. *wmcs*, 0:70–78, 2005.

[82] Paul Madsen. Federated identity and web services. *Information Security Technical Report*, 9:56–65, 2004. Information Security Technical Report. Vol. 9, No. 3.

[83] Bertrand Meyer. The grand challenge of trusted components. In *ICSE '03: Proceedings of the 25th International Conference on Software Engineering*, pages 660–667, Washington, DC, USA, 2003. IEEE Computer Society.

[84] Sun Microsystems. About the java technolog. http://java.sun.com/docs/books/tutorial/getstarted/intro/definition.html. Available Online, 2007.

[85] Cameron Morris. Browser based trust negotiation. Master's thesis, Brigham Young University, 2006.

[86] BBC News. Paypal introduces security token. http://news.bbc.co.uk/1/hi/technology/6357835.stm. Available Online, February 2007.

[87] Zeltser L. Inters S. Kent Frederick K. Ritchey R.W. Northcutt, S. *Inside networks perimeter security*. New Riders Publishing, Illinouis, U.S.A., 2003.

[88] Graham Palmer. De-perimeterisation: Benefits and limitations. *Information Security Technical Report*, 10(4):189–203, 2005.

[89] Chengyuan Peng. Gsm and gprs security. Seminar on Network Security, 2000.

[90] A Pfitzmann and M Waidner. Networks without user observability - design options. pages 245–253, 1986.

[91] B. Pfitzmann and M. Backes. Federated identity-management protocols - where user authentication protocols may go. In *11th International Workshop on Security Protocols*, volume 3364 of *Lecture Notes in Computer Science*, pages 153–174. Springer-Verlag, 2003. Springer-Verlag, Berlin Germany, 2003. copyright Springer-Verlag, 2005.

[92] Birgit Pfitzmann. Privacy in enterprise identity federation - policies for liberty 2 single sign on. *Information Security Technical Report*, 9(1):45–58, 2004.

[93] Birgit Pfitzmann and Michael Waidner. Privacy in browser-based attribute exchange. In *WPES '02: Proceedings of the 2002 ACM workshop on Privacy in the Electronic Society*, pages 52–62, New York, NY, USA, 2002. ACM Press.

[94] M.G. Reed, P.F. Syverson, and D.M. Goldschlag. Anonymous connections and onion routing. *IEEE Journal on selected areas in Communication*, 16(4):482–494, May 1998.

[95] Michael K. Reiter and Aviel D. Rubin. Anonymous web transactions with crowds. *Commun. ACM*, 42(2):32–48, 1999.

[96] D.W. Robinson. Defence in depth. Technical report, SANS Institute, Security Reading Room., 2002.

[97] Alan Rodger. Access management: The key to compliance. *Card Technology Today*, pages 11–12, September 2004.

[98] T. Ryutov and C. Neuman. The specification and enforcement of advanced security policies. In *POLICY '02: Proceedings of the 3rd International Workshop on Policies for Distributed Systems and Networks (POLICY'02)*, page 128, Washington, DC, USA, 2002. IEEE Computer Society.

[99] Tatyana Ryutov, Clifford Neuman, Dongho Kim, and Li Zhou. Integrated access control and intrusion detection for web servers. In *ICDCS '03: Proceedings of the 23rd International Conference on Distributed Computing Systems*, page 394, Washington, DC, USA, 2003. IEEE Computer Society.

[100] Tatyana Ryutov, Li Zhou, Clifford Neuman, Travis Leithead, and Kent E. Seamons. Adaptive trust negotiation and access control. In *SACMAT '05: Proceedings of the tenth ACM*

symposium on Access control models and technologies, pages 139–146, New York, NY, USA, 2005. ACM Press.

[101] Christian Schlager, Thomas Nowey, and Jose A. Montenegro. A reference model for authentication and authorisation infrastructures respecting privacy and flexibility in b2c ecommerce. In *ARES '06: Proceedings of the First International Conference on Availability, Reliability and Security (ARES'06)*, pages 709–716, Washington, DC, USA, 2006. IEEE Computer Society.

[102] Bruce Schneier. *Secrets & Lies: Digital Security in a Networked World.* John Wiley & Sons, Inc., 2000.

[103] K. Seamons, M. Winslett, T. Yu, B. Smith, E. Child, J. Jacobson, H. Mills, and L. Yu. Requirements for policy languages for trust negotiation. In *POLICY '02: Proceedings of the 3rd International Workshop on Policies for Distributed Systems and Networks (POLICY'02)*, page 68, Washington, DC, USA, 2002. IEEE Computer Society.

[104] RSA Security. Rsa mobile and rsa securid. http://www.rsa.com. Available Online, December 2002.

[105] J. Snyder. 6 strategies for defence-in-depth. Opus White Paper., 2004.

[106] Internet Society. http://www.isoc.org. Available Online, 2007.

[107] Markus A. Stadler, Jean-Marc Piveteau, and Jan L. Camenisch. Fair blind signatures. *Lecture Notes in Computer Science*, 921:209–219, 1995.

[108] Whitley R. Stamp, P. Jericho forum looks to bring network walls tumbling down. Available Online, 2005.

[109] Gary Stoneburner, Alice Goguen, and Alexis Feringa. Risk management guide for information technology systems. Technical report, National Institue of Standards and Technology, 2002.

[110] S.R. Subramanya and B.K. Yi. Mobile communications - an overview. *IEEE Potentials*, 24(5):36– 40, December 2005.

[111] ETSI TC-SMG. Specification of the subscriber identity module - mobile equipment (sim - me) interface (gsm 11.11). Technical Specification Version 5.0, European Telecommunications Standards Institute, December 1995. GSM 11.11 Version 5.0.0 December 1995.

[112] Douglas Thain. Identity boxing: A new technique for consistent global identity. In *SC '05: Proceedings of the 2005 ACM/IEEE conference on Supercomputing*, page 51, Washington, DC, USA, 2005. IEEE Computer Society.

[113] Jie Tian, Liang Li, and Xin Yang. Fingerprint-based identity authentication and digital media protection in network environment. *Journal of Computer Science and Technology*, 21(5):861 –870, 06 2006.

[114] J. Tiller. Security virtues of a common infrastructure. international network services. Cisco White Paper, 2005.

[115] Doug Turner and Ian Oeschger. *Creating XPCOM Components.* Brownhen Publishing, 2003.

[116] Wen-Guey Tzeng. A secure system for data access based on anonymous authentication and time-dependent hierarchical keys. In *ASIACCS '06: Proceedings of the 2006 ACM Symposium on Information, computer and communications security*, pages 223–230, New York, NY, USA, 2006. ACM Press.

[117] Pieter Ben van der Merwe. Mobile commerce over gsm: A banking perspective on security. Master's thesis, University of Pretoria, October 2003.

[118] Dr. Do van Thanh. Offering sim strong authentication to internet services. White Paper, October 2006.

[119] Ray Wagner and Gregg Kreizman. Frequently asked questions about federated identity. Technical Report G00139097, Gartner Research, April 2006.

[120] L Wilkes. Roi - the costs and benefits of web services and service oriented architecture. http://roadmap.cbdiforum.com/reports/roi/, accessed november 2006. Available Online, 2003.

[121] K. Wilson. de jure, de facto. http://www.bartleby.com/68/7/1707.html. Available Online, 1993.

[122] Phillip Windley. *Digital Identity.* O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472, first edition, August 2005. isbn: 0-596-00878-3.

[123] Marianne Winslett, Ting Yu, Kent E. Seamons, Adam Hess, Jared Jacobson, Ryan Jarvis, Bryan Smith, and Lina Yu. Negotiating trust on the web. *IEEE Internet Computing*, 06(6):30–37, 2002.

[124] JoAnne Woodcock. The internet and the web. http://technet.microsoft.com/en-us/library/bb726945.aspx. Available Online, September 2007.

[125] The World Wide Worx. http://www.theworx.biz/access05.htm. Available Online, March 2007.

[126] Min Wu. *Fighting Phishing at the User Interface.* PhD thesis, Massachusetts Institute of Technology, August 2006.

[127] Christos Xenakis and Lazaros Merakos. Security in third generation mobile networks. *Computer Communications*, 27(7):638–650, May 2004.

[128] R. Yavatkar, D. Pendarakis, and R. Guerin. A framework for policy-based admission control. RFC: 2753 Category: Informational 2753, Network Working Group, The Internet Society, 2000.

Chapter 8

Appendix

8.1 Glossary

2G second generation

3G third generation

De-P de-perimeterisation

GPRS general packet radio services

GSM global system for mobile communication

HLR home location register

HTTP hypertext transfer protocol

IMSI international mobile subscriber identity

IMEI international mobile equipment identity

IP internet protocol

IPsec IP security

MAC message authentication code

PIN personal identity number

6 April 2009

600M4187

Mr SG Miles
P O Box 2684
Beacon Bay
East London
5205

Dear Mr Miles

I acknowledge receipt of the library copies of your thesis.

It is with much pleasure that I am now able to inform you that the award to you of the degree of Master of Science has been approved by the Dean of Science acting on behalf of Faculty.

Details of the 2009 graduation ceremonies are available for information on our web page (http://www.ru.ac.za/registrar/academicadministration/graduation/). The Graduation info booklet and reply card will be available on the site from late December.

Please note that you may only use the appropriate letters for your qualification after your name once you have graduated either *in praesentia* or *in absentia.*

Congratulations and every best wish for continued success in the future.

Yours sincerely

Dr Stephen Fourie
REGISTRAR

cc Dean of Science
Head of Department of Computer Science
Mr B Irwin

Donna Griffin

From:	Donna Griffin [d.l.Griffin@ru.ac.za]
Sent:	07 April 2009 09:28 PM
To:	'SHAUN.G.MILES@gmail.com'
Cc:	'r.Bernard@ru.ac.za'; 'b.irwin@ru.ac.za'; 'p.wentworth@ru.ac.za'
Subject:	Loose leaf copies

600M4187

Dear Mr Miles

I acknowledge receipt of the library copies of your thesis.

It is with much pleasure that I am now able to inform you that the award to you of the degree of Master of Science has been approved by the Dean of Science acting on behalf of Faculty.

Details of the 2009 graduation ceremonies are available for information on our web page (http://www.ru.ac.za/registrar/academicadministration/graduation/). The Graduation info booklet and reply card will be available on the site from late December.

Please note that you may only use the appropriate letters for your qualification after your name once you have graduated either *in praesentia* or *in absentia*.

Congratulations and every best wish for continued success in the future.

Yours sincerely

Dr Stephen Fourie
REGISTRAR

cc Dean of Science
Head of Department of Computer Science
Mr B Irwin

DISCLAIMER: *The Registrar's Division at Rhodes University uses e-mail as the primary means of communication and the electronic version of this letter has official status. Should a paper copy be required, please contact academicadmin@ru.ac.za.*

Donna-lee Griffin
Administrative Officer
Fax: +2746 6038127
Internal tele: x 8216

Academic Administration
Registrar's Division, RU
Eden Grove Building
Lucas Avenue
Grahamstown RSA
6139

RHODES UNIVERSITY
Grahamstown • 6140 • South Africa

COMPUTER SCIENCE DEPARTMENT • Tel: (046) 603 8626 • Fax: (046) 636 1915 • e-mail: b.irwin@ru.ac.za

Tuesday 31 March 2008

The Registrar
Rhodes University
Grahamstown

RE: Submission of Thesis by SG MILES (00M4187)

This letter serves to advise that Mr MILES has completed the corrections of his Masters thesis titled *"An Investigation of Issues of Privacy, Anonymity and Multi-Factor Authentication in an Open Environment"* as specified by his examiners. These corrections have been completed to my satisfaction.

Regards

Barry Irwin
MSc (Rhodes) CISSP

RHODES UNIVERSITY
Where leaders learn

Thesis Final Submission
(Print & Electronic)

To be completed by ALL students submitting a thesis. Please type or write clearly in BLOCK LETTERS.

You are encouraged to submit an electronic version of your thesis in PDF format for archiving in the ReRR (Rhodes eResearch Repository) http://eprints.ru.ac.za/ to enable world-wide access to your research.

PLEASE SEND THE FORM TOGETHER WITH THE PRINT AND OPTIONAL ELECTRONIC COPIES TO THE REGISTRAR'S DIVISION

A. STUDENT INFORMATION

Student Name: SHAUN GRAEME MILES
First name *Middle name or initial* *Surname*

Student number: g00M4187 **Phone :** 072 027 9557

Email address(es): SHAUN.G.MILES@gmail.com

May your e-mail address be made available on the ReRR web site? *Yes* / [✓ *No*]

B. THESIS INFORMATION

Degree: M.Sc. **Graduation date (MM/YY):** 04/09

Department: Computer Science **Faculty:** Science

Supervisor: Barry Irwin **Co-supervisor:**

Title of thesis: AN INVESTIGATION of ISSUES of Privacy, Anonymity and Multi-Factor authentication in An Open Environment

Suggested keywords: Privacy, Anonymity, Multi-Factor Authentication

I hereby submit :
☑ 2 unbound print copies ; and
☑ electronic copy on 1 (number of) □ CDs, □ DVDs, or □ other

(CDs/DVDs should be labeled clearly with your name, degree, dept and date of graduation)

I declare that:
☑ all the print & electronic copies submitted are full and final versions of my thesis, i.e. the same as the final copy approved by the examiners;
☑ I have checked that the electronic copy is identical to the print copy;
☑ I have checked the disc(s) for corruption.

C. STUDENT AGREEMENT

I agree to have the electronic version of my thesis placed in the Rhodes eResearch Repository http://eprints.ru.ac.za with the following status (*choose one)* :

☑ 1. Release the entire work immediately for worldwide access

□ 2. Delay release of the entire work for a **holding period** of

□ 1 year, or

□ 2 years

After the holding period automatically release the work for worldwide access.
NOTE: Embargoes of print copies require the approval of Senate.

I hereby certify that, if appropriate, I have obtained and attached hereto a written permission statement from the owner(s) of any **third-party copyrighted material** in my thesis, for example visual images .

I hereby grant to Rhodes University and its agents the non-exclusive license to archive and make accessible, under the conditions specified below, the e-version of my thesis in whole in all forms of media, now or hereafter known. I retain all ownership rights to the copyright of the thesis. I also retain the right to use in future works (such as articles or books) all or part of this thesis.

I agree to abide by the statements above, and agree that this Submission Form updates any and all previous *forms submitted heretofore.*

SIGNATURE OF STUDENT : ______________________________ DATE: 31/01/09

D. SUPERVISOR AGREEMENT

Name of supervisor: B. V. W Irwin.

May your e-mail address be made available on the ReRR web site, and if so, please supply your e-mail address? [✓ Yes] / No ; **E-mail address:** b.irwin@ru.ac.za.

I hereby verify that all the necessary changes as requested/indicated by the examiners have been made and am satisfied that this copy is the final copy.

I hereby verify that I have reviewed the final electronic version of the document to be submitted and have determined that it is an accurate representation of the thesis.

SIGNATURE OF SUPERVISOR: ______________________________

DATE : 31/03/2009.

E. TO BE COMPLETED BY THE REGISTRAR'S DIVISION

☑ I am satisfied that ALL sections A - D have been completed.

The student has provided the Registrar's Division with

☑ 2 unbound print copies ; and

☑ electronic copy on 1 (number of) [✓ CDs]/DVDs

SIGNATURE OF OFFICER : ______________________________

PRINTED NAME : NTOSH RAOU DATE: 22/07/09

Thesis corrections:

* Technical revision:
Fixed all typo, grammar and spelling errors as specified by exmaniners and those found subsequently.

* Streamlined text heavy Chp 2:
Removed a couple of paragraphs that did not fit in with the direction of the research.

* Add substence to Chp 6&7:
Tightened up some of the paragraphs, included more references to previous chapters.
Added an extra dimension of the perspective of the attacker to the scenario discussion.
Fleshed out the scenarios in more detail.
Tidied up the conclusion, brought the different sections together better.
Added future work section to chp 7 (which wasnt in print).

Shaun Miles 31/03/09

Miles

SIM subscriber identity module

SAT SIM application toolkit

USSD unstructured supplementary services data

XPCOM cross platform component model