WSP^3: A Web Service Model for Personal Privacy Protection

by

Jacobus Albertus Ophoff

WSP^3: A Web Service Model for Personal Privacy Protection

by

Jacobus Albertus Ophoff

Dissertation

submitted in fulfillment
of the requirements
for the degree

Magister Technologiae

in

Information Technology

in the

Faculty of Computer Studies

of the

Port Elizabeth Technikon

Promoter: Prof. Reinhardt A. Botha

December 2003

Declaration

I, Jacobus Albertus Ophoff, hereby declare that:

- The work in this dissertation is my own work.
- All sources used or referred to have been documented and recognized.
- This dissertation has not previously been submitted in full or partial fulfillment of the requirements for an equivalent or higher qualification at any other recognized educational institute.

Jacobus Albertus Ophoff

Abstract

The prevalent use of the Internet not only brings with it numerous advantages, but also some drawbacks. The biggest of these problems is the threat to the individual's personal privacy. This privacy issue is playing a growing role with respect to technological advancements. While new service-based technologies are considerably increasing the scope of information flow, the cost is a loss of control over personal information and therefore privacy. Existing privacy protection measures might fail to provide effective privacy protection in these new environments.

This dissertation focuses on the use of new technologies to improve the levels of personal privacy. In this regard the WSP^3 (Web Service Model for Personal Privacy Protection) model is formulated. This model proposes a privacy protection scheme using Web Services. Having received tremendous industry backing, Web Services is a very topical technology, promising much in the evolution of the Internet.

In our society privacy is highly valued and a very important issue. Protecting personal privacy in environments using new technologies is crucial for their future success. These facts, combined with the detail that the WSP^3 model focusses on Web Service environments, lead to the following realizations for the model:

- The WSP^3 model provides users with control over their personal information and allows them to express their desired level of privacy. Parties requiring access to a user's information are explicitly defined by the user, as well as the information available to them.

- The WSP^3 model utilizes a Web Services architecture to provide privacy protection. In addition, it integrates security techniques, such as cryptography, into the architecture as required.

- The WSP^3 model integrates with current standards to maintain their benefits. This allows the implementation of the model in any environment supporting these base technologies.

In addition, the research involves the development of a prototype according to the model. This prototype serves to present a proof-of-concept by illustrating the WSP^3 model and all the technologies involved.

The WSP^3 model gives users control over their privacy and allows everyone to decide their own level of protection. By incorporating Web Services, the model also shows how new technologies can be used to offer solutions to existing problem areas.

Acknowledgements

This work would not have been possible alone. I would like to thank the following people for their contribution to this study:

- My family, for their love, motivation and encouragement.
- Madri, for her love and support.
- My promotor, Professor Reinhardt Botha, for his guidance, patience and supervision.
- My colleagues at the Port Elizabeth Technikon for their assistance.
- The National Research Foundation for their financial assistance.
- God, for giving me the ability and opportunity to do this work.

The financial assistance of the Department of Labour (DoL) towards this research is hereby acknowledged. Opinions expressed and conclusions arrived at, are those of the author and are not necessarily to be attributed to the DoL.

Contents

List of Figures

Part I

Background

Chapter 1

Introduction

The widespread adoption of the Internet by millions of users worldwide has considerably increased the scope of information flow. However, this has also increased the risk to individuals' personal privacy over the last decade. Concern has grown over private information collection and to what extent individual control over that information can be maintained.

"Our awareness has been raised; our resistance, increased. We must act with vigor and vigilance to ensure our privacy, for it is a right that affects every individual with an identity and a personal history" (Eder, 1994).

To encourage the use of privacy policies several market leaders, including IBM and Microsoft, have announced that they will no longer advertise on Web sites that do not post privacy policies (Swire, 2000).

According to Henderson and Snyder (1999) another step to improve privacy protection has been to propose self-regulatory policies and procedures. The Association for Computing Machinery has included a section on privacy in its Code of Ethics and Professional Conduct. Their code states that "It is the responsibility of professionals to maintain the privacy and integrity of data describing individuals. This includes taking precautions to ensure the accuracy of data, as well as protecting it from unauthorized access or accidental disclosure to inappropriate individuals" (ACM, 1992).

Although privacy policies have become more common on Web sites around the Internet, there are still growing concerns about corporate use of personal information for profiling individuals, and continuing controversy about the effectiveness of industry self-regulation (Cranor, 2000).

Freedman (1987, p. 93–94) has made the observation that highly sophis-

ticated technology with its enhanced capacity for communication, computation, storage, and retrieval has given personal information privacy and the right of privacy new meaning.

New service-based technologies are making it increasingly easier for organizations to pull information from a number of external sources. It has become relatively easy and cheap to cross-correlate and share information through the Internet. The price for these capabilities comes at a high cost to personal information – control over personal information is lost and the threat to an individual's privacy increased (Henderson & Snyder, 1999).

This study is therefore primarily motivated by the realization that the individual's right to privacy needs to be protected.

1.1 Motivation For This Study

In addition to the aforementioned comments, several other realizations motivate and support this study.

1.1.1 The Realization That Personal Privacy Is Important

Privacy has been defined as the right of individuals to control the collection and use of personal information about themselves. The right to privacy has become one of the most important ethical issues of the information age (Mason, 1986).

Privacy is a fundamental right recognized in the United Nations Universal Declaration of Human Rights, the Council of Europe's Convention for the Protection of Individuals with Regard to Automatic Processing of Personal Data, the International Covenant on Civil and Political Rights, and many other international and regional treaties.

According to Privacilla (2001a) a person has privacy when two factors are in place. The person must have the ability to control information about him- or herself, and must exercise that control consistent with his or her values.

Privacy must not be confused with confidentiality which refers to keeping sensitive information secret and protected from inappropriate viewing. Privacy requires that the confidentiality of user information is protected both

in transit and in storage (Madsen & Adams, 2002).

Ultimately privacy, as with trust, is reliant on our perception of it. It is not necessarily important how private or safe we are (although this is a vital component) but whether we perceive ourselves to be safe and private. Over recent years the importance of the users' perceptions has been identified with a move to increase perceptions of privacy (Adams, 2000).

1.1.2 The Realization That Privacy Protection Is Becoming Crucial

According to the Center for Democracy and Technology (CDT) (2000) privacy is the number one concern of Internet users; it is also the main reason why people still avoid the Internet. Survey after survey indicates growing concern (Cranor, Reagle, & Ackerman, 1999). While both private and government intrusions threaten privacy, the existing diverse patchwork of privacy laws and practices fails to provide comprehensive protection. Instead these laws and practises cause confusion, fuelling a sense of distrust and scepticism, limiting realization of the Internet's potential.

Information privacy issues remained at the forefront of consumer concern in the 1990's. This attention has been brought about by the increasing impact of IT on daily life (Smith, 1993) and by recent media attention. As evidence of the renewed interest, articles have appeared in newspapers and magazines. Television shows have included exposés involving loss of personal data, and books have been published on privacy. All have contributed to the growing concern about information privacy (Henderson & Snyder, 1999).

According to Madsen and Adams (2002) some of the reasons why privacy is receiving so much attention include the following:

- The unfamiliarity most people have with the technologies that make up the Internet. Users are asked to make decisions on issues of which they have no knowledge.

- Both authorized and unauthorized information access has become easier due to the interconnectedness of networks. Information is now stored in a database on a Web server instead of a separate off-line machine or filing cabinet.

- The emergence of mobile technologies will enable scenarios unimaginable in the past, such as determining the user's exact location through the signals emitted by mobile devices.

- Federated identity such as Microsoft .NET My Services and the Liberty Alliance, which allows information to be shared between Web sites, has amplified concern about the misuse of that information.

These issues need to be addressed and Internet users educated before privacy fears will be subdued.

1.1.3 The Realization That Web Services Are Revolutionary

Web Services have been much talked about in the IT industry over the past couple of years. This new breed of Web application is promising to take the Web to its next stage of evolution.

According to Kirtland (2001a), "a Web Service is programmable application logic accessible using standard Internet protocols. Web Services combine the best aspects of component-based development and the Web."

Glass (2000) gives the following description: "A Web service is a collection of functions that are packaged as a single entity and published to the network for use by other programs. Web services are building blocks for creating open distributed systems, and allow companies and individuals to quickly and cheaply make their digital assets available worldwide."

One of the biggest problems on the Internet today is allowing information to be shared between applications. As a solution to this problem, Web Services establish a method of standardizing communication, making information sharing between applications and devices across the Internet easier (Coursey, 2002).

To do this a typical Web Services architecture consists of three components:

- A service provider who creates services and publishes them for use by registering the services with a service broker.

- A service broker who maintains a registry of published services.

- A service requester who finds services by searching the broker's registry and then binds his/her application to the provider to use particular services.

Figure 1.1 illustrates the interaction between provider, broker and requester in the publication, discovery and consumption of Web Services (Roy & Ramanujan, 2001).

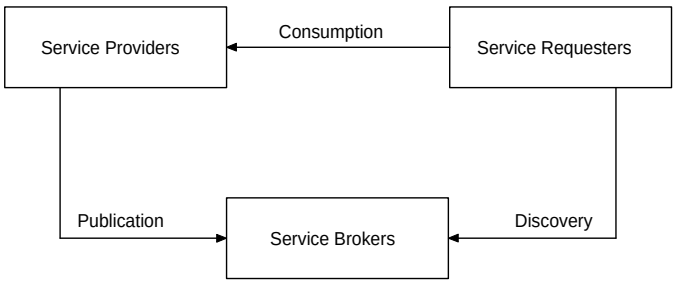

Figure 1.1: Web Services Architecture

In a typical scenario a service provider would develop and host a Web Service for public use and publish details about the service to a broker. The broker provides a convenient repository for clients to discover services they require. After a client discovers such a service it is then invoked from the Web Service host. By providing a standardized method for publishing services and discovering them, Web Services allow applications to be created with greater efficiency.

Despite all the promise Web Services hold, there are still a few concerns that have been raised. One of the biggest has been the lack of security. "Web services have an easy security model: They don't have one" (Vaughan-Nichols, 2001). Additionally, Simple Object Access Protocol (SOAP), the main Web Services protocol, is written in Extensible Markup Language (XML), which is written in clear text, making it very easy to read.

Another concern is that the Web Services model is based on a Web page that uses third-party services, often without the knowledge of the user.

User privacy is a problematic issue for developers of Web Services and the applications that use them. A lot of work has to be done beyond implementing the core functionality of your service if personally identifiable information needs to be maintained. All five elements of fair information practises need to be addressed: notice, consent, access, security and enforcement. You will

need to determine when you must address these directly with users, and when you can defer the privacy issues to the applications using your Web Service (Kirtland, 2001b).

1.2 Problem Statement

With Internet users demanding ever increasing privacy protection measures, it is essential that new technologies address this need adequately. The problem the author addresses in this study is whether Web Services can be exploited to provide effective personal privacy protection.

In order to address this problem successfully a number of sub-problems need to be addressed:

- What level of personal privacy protection is needed by an individual?
- How can Web Services be leveraged for privacy protection?
- How can privacy be protected in mainly insecure supporting technologies such as the Extensible Markup Language (XML)?

Effectively answering the above problems would ensure that the following objectives are achieved.

1.3 Objectives

The principal objective of this study is to develop a service-based model, called WSP^3, to protect the personal privacy of individuals on the Internet. This model will focus on effective privacy support in a Web Services environment.

To successfully implement such a model a number of sub-objectives need to be addressed:

- The individual should be allowed to specify the level of privacy of his/her information.
- Privacy protection must be implemented using Web Services and related technologies.

- The model must integrate seamlessly with current standards, such as XML.

These objectives will be achieved through the following methodology.

1.4 Methodology

A thorough literature study will form the basis of the project. Firstly, an extensive study regarding online privacy will be done. This study will consider the current state of privacy on the Internet and the implications for service-based environments. Secondly, a study into service-based environments with specific focus on Web Services will be done. This will consider the current state of Web Services, the underlying technologies such as SOAP and XML, and the current privacy mechanism's implementations.

Using the knowledge gathered during this study a model for introducing privacy protection into service-based environments will be developed. The proposed model will be demonstrated through a prototype implementation. This will illustrate the viability of the model.

The results of this study would be reported in the form of an academic paper and a dissertation.

1.5 Layout Of Dissertation

The proposed layout of the dissertation is depicted in Figure 1.2 and is divided into three parts:

Part I introduces the domain of discourse to the reader and is divided into four chapters. **Chapter 1** provides some background into the problem area in order to delineate the problem. Next, **Chapter 2** provides an overview of privacy and current privacy legislation. In **Chapter 3** the existing privacy protection mechanisms are discussed. **Chapter 4** focuses on the Web Services architecture and its underlying technologies.

Part II is dedicated to the development of the proposed model. In **Chapter 5** the conceptual foundation of the proposed model is provided. **Chapters 6 and 7** develops the details of the model in depth. Thereafter, **Chapter 8** demonstrates the viability of the model via the development of a prototype.

Part III contains the epilogue. **Chapter 9** will conclude the dissertation and suggest areas for further research.

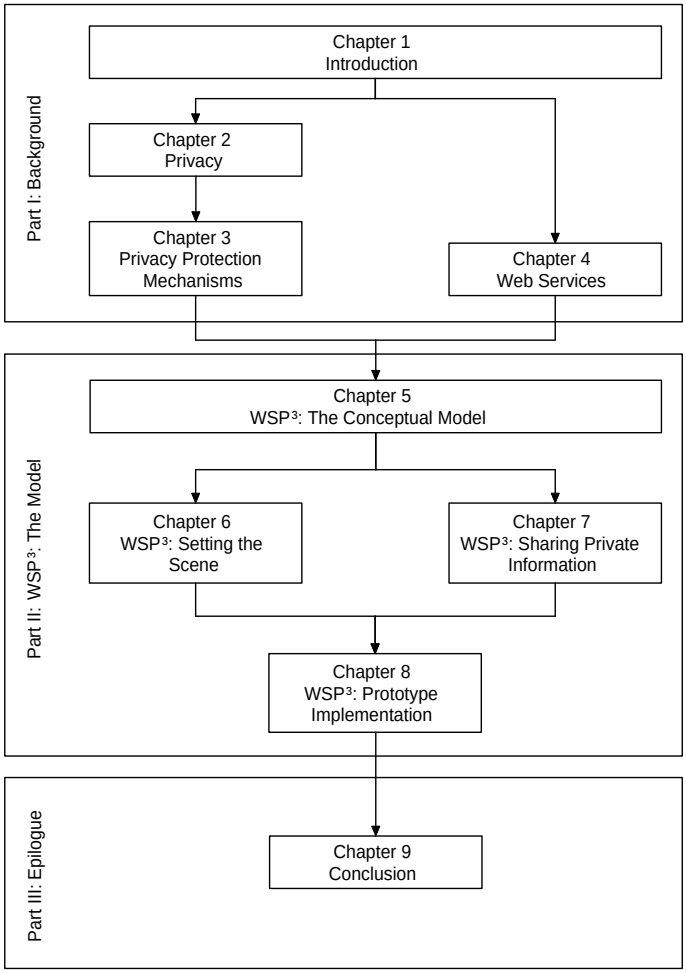

Figure 1.2: Layout Of Dissertation

Chapter 2

Privacy

"Privacy is perhaps the most discussed ethical and social issue in computer and information technology" (Weckert & Adeney, 1997, p. 75). People are worried about the ease with which personal information can be collected, how it is used and shared, and to what extent they will have access to it. It is clear that we live in a society that values personal privacy and is concerned about invasions into privacy (Belsey & Chadwick, 1992, p. 77).

This chapter provides a brief examination of the nature of privacy, the most important issues and concepts surrounding it and an overview of current privacy legislation.

2.1 The Nature Of Privacy

Privacy is a concept that is normally regarded to be valuable and a good thing to have. However, after careful consideration it is not easy to explain what privacy is and why it is so valuable. Privacy is an illusive concept, entailing different issues for different individuals. Because it is an abstract and debatable issue a lot of people use the term without knowing what it really means. This section will look at general questions surrounding privacy and people's attitude towards it.

2.1.1 Privacy Definition

The term "privacy" is used in many different ways and does not always mean the same thing. Privacy is typically categorized into several areas of everyday life.

According to Belsey and Chadwick (1992, p. 83) there are three overlapping areas of personal life where the protection of privacy might be required. These are bodily or physical privacy, mental or communicational privacy and informational privacy. The last area is of particular relevance to this study and is further explained as providing "... protection for personal information which is legitimately held in the files of public and private organizations, and prevents the disclosure of such information to third parties. 'Legitimately' not only means in accordance with the law but also with the subject's knowledge and consent."

The Privacilla (2001b) group states that privacy is a condition people maintain by controlling who receives information about them and the terms on which they receive it. Importantly, it is a subjective, personal condition. One person cannot decide for another what their sense of privacy should be. They continue to say that a person has privacy when two factors are in place. Firstly, people must have the ability to control information about themselves and, secondly, they must exercise that control in accordance with their values.

According to The Privacy Hub (2002), privacy allows individuals to decide about the processing of their personal data and the protection of their intimate sphere. Privacy concerns can be generalized into four categories:

- Improper acquisition of personal information, including its access, collection and distribution.
- Improper use of information, including its deployment for reasons other than which it was explicitly collected and its transfer to other parties.
- Privacy invasion, including unwanted solicitation of personal data.
- Improper storage of information.

Finally, TechTarget (2001) stated that Internet privacy can be divided into the following concerns:

- What personal information can be shared with whom.
- Whether messages can be exchanged without anyone else seeing them.
- Whether and how one can send messages anonymously.

From these definitions and concerns it can be seen that privacy has many important factors often influencing more of our lives than we are aware of.

2.1.2 Privacy As A Right

The value of privacy is deeply rooted in history. "The Bible has numerous references to privacy. Jewish law has long recognized the concept of being free from being watched. There were also protections in Classical Greece and ancient China" (EPIC and Privacy International, 2000b).

Privacy protection has existed in western countries for hundreds of years. In 1361 the "Justices of the Peace Act" provided for the arrest of peeping toms and eavesdroppers in England. In the centuries following, various countries developed specific privacy protections. In 1776 the Swedish parliament passed the "Access to Public Records Act", requiring all government-held information to be used for legitimate purposes. In 1858 France prohibited the publication of private facts and in 1889 the Norwegian criminal code forbade the publication of personal or domestic information (EPIC and Privacy International, 2000b).

The modern benchmark at international level can be found in the Universal Declaration of Human Rights (1948) which states that "No one should be subjected to arbitrary interference with his privacy, family, home or correspondence, nor to attacks on his honor or reputation. Everyone has the right to the protection of the law against such interferences or attacks".

Today numerous international human rights treaties specifically recognize privacy as a right. There is a right to privacy because it does (or can) protect individuals from various kinds of harm (Weckert & Adeney, 1997, p. 77–78). Living in a society where privacy is continuously abused or non-existent would be intolerable (Belsey & Chadwick, 1992, p. 78).

2.1.3 Privacy On The Internet

Privacy is the number one concern of Internet users today. People still avoid the Internet because of the lack of protection for individuals and this is preventing the Internet from realizing its full potential. (Center for Democracy and Technology (CDT), 2000)

Users' attitude towards privacy is often different on the Internet than in

real life. A survey by Cranor et al. (1999) produced the following major findings about user's attitude towards Internet privacy:

- Internet users are more likely to provide information when they are not identified.
- Some types of data are more sensitive than others, an example being a telephone number versus an email address.
- Many factors are important in decisions about information disclosure, the most important factor being whether or not information will be shared with other companies and organizations.
- Acceptance of the use of persistent identifiers (eg. cookies) varies according to their purpose.
- Internet users dislike automatic data transfer and most do not want tools to transfer information about them to Web sites automatically.
- Internet users dislike unsolicited communication and indicated a strong desire to avoid unsolicited communication resulting from providing information to Web sites.
- A joint program of privacy policies and privacy seals seemingly provides a comparable level of user confidence as that provided by privacy laws.

From this survey Cranor et al. (1999) concluded that suitable interfaces for privacy protocols will be a major issue and that "it seems unlikely that a one-size-fits-all approach to online privacy is likely to succeed".

From the above discussion it can be concluded that privacy is not a simple concept and that there is no straightforward implementation. Internet privacy is an area in which users still have a lot of fears and concerns. As such it is important that a clear articulation is formed of what privacy really is, before any protection mechanisms can be defined.

2.2 Privacy Issues And Concepts

To gain a comprehensive insight into privacy a number of concepts and issues central to understanding privacy need to be addressed. Figure 2.1 presents an overview of these concepts and issues.

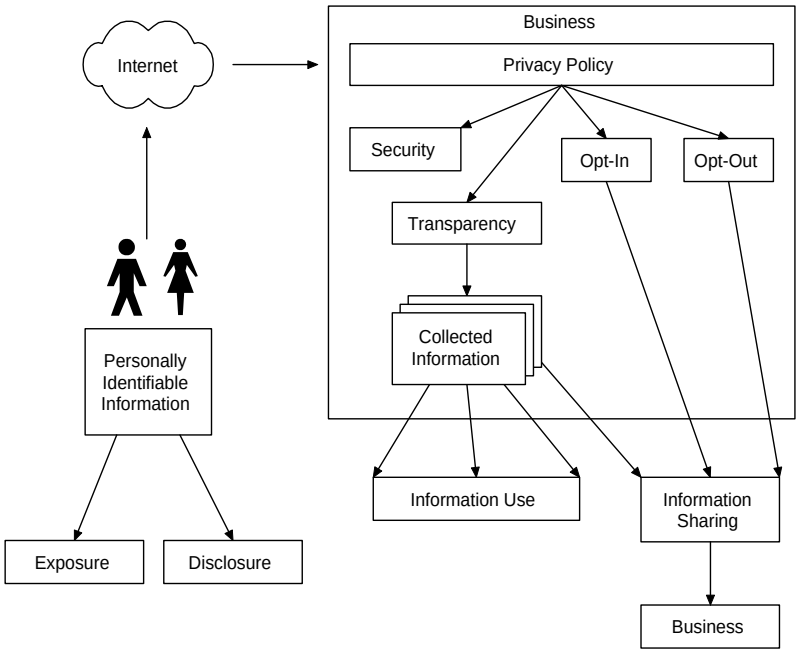

Figure 2.1: Privacy Concepts

The following paragraphs will examine various aspects of privacy regarding the individual, as indicated in the figure. Firstly, identifying information about an individual will be discussed as well as the related concepts of exposure and disclosure of such information. Thereafter, issues relating to interaction with a business are examined. These include the existence of a privacy policy, security measures to enforce such a policy and the strategies by which businesses collect customer information, referred to as opt-in or opt-out. Next, the access consumers have to their data (transparency) is discussed. Finally, the use and sharing of collected information is considered.

2.2.1 Personally Identifiable Information

Personally identifiable information is information that is unique to an individual and can be used to identify an individual or distinguish that individual from many (or all) others. Examples of personally identifiable information are an identity number, a telephone number, and possibly an email address. Data that does not uniquely identify the bearer, such as age, gender and salary, are not typically considered as personally identifiable information. Although less concerning than personally identifiable information, such anony-

mous data can be very relevant to privacy if it can be linked to personally identifiable information.

2.2.2 Exposure And Disclosure

Although the concepts of exposure and disclosure are distinct, they are both related to privacy. Exposure deals with identity: "Am I willing to reveal who I am to other entities within a certain context?" Disclosure has to do with other information about an individual: "Am I willing to reveal personal or sensitive information to other entities for some purpose?" Although these concepts are closely related it is useful to keep them separate because in many environments an authentication step (which may expose identity) occurs prior to the remainder of the transaction that discloses additional information (Madsen & Adams, 2002).

Exposure can be further categorized into techniques providing anonymity, pseudonymity or veronymity:

- Anonymity ("no name") refers to the use of no name or a name that has never been used before and will never be used again for a transaction. Its defining property is that no linkage is possible between the transaction and the entity performing the transaction, and no linkage between transactions can be done.

- Pseudonymity ("false name") refers to the use of a particular name for multiple transactions which is different than the real name of the entity performing the transactions. Thus the defining property of pseudonymity is that no linkage is possible between the transaction and the entity performing the transaction, but a linkage is possible between different transactions through the pseudonyms.

- Veronymity ("true name") refers to the use of the real name of the entity performing the transaction. Linkage between the transaction and entity and between transactions performed by the same entity is possible.

Consequently identity information may not be exposed at all (anonymous transactions), may be partly exposed (pseudonymous transactions) or may be fully exposed (veronymous transactions).

2.2.3 Privacy Policy

It is essential for a business to maintain a high level of trust with consumers by publishing the company's policies for data protection and secondary data sharing. Even though most consumers may never read these policies, not presenting the option may convince the consumer to do business elsewhere. Privacy policies typically detail the protection level and mechanisms employed by the company, as well as when, why, how and with whom data in their possession will be shared (Smith, 1993).

2.2.4 Security

Security and privacy are very closely related. A privacy policy is worthless if there are no security mechanisms in place to enforce it. Security encompasses many concepts relevant to privacy such as:

- Confidentiality relates to keeping sensitive information secret and protected from inappropriate access. Privacy requires that information confidentiality be protected both in transit and storage.
- Authorization is the process of determining what an individual or business entity is allowed to do. An example might be allowing one individual to view your calendar and another to write to it.
- Non-repudiation deals with preventing entities from denying an action of theirs. This is relevant to privacy because it would prevent a business from breaking its privacy policy and denying its actions later on.
- Authentication refers to proving that individuals or businesses are who they claim to be.

2.2.5 Opt-In Versus Opt-Out

Businesses generally use one of two strategies to get the approval of consumers for sharing their information. These two strategies, known as Opt-in and Opt-out, differ in the assumptions they make about the value of the data and what the appropriate default rule for sharing should be (Bellman, Johnson, & Lohse, 2001).

The opt-in strategy assumes that consumer information has high value and gives the consumer an explicit choice to approve the sharing of his/her data as opportunities arise. The default with the opt-in strategy is not to share consumer information, which is consistent with the assumption of valuable information. If consumers agree to share their information, they must explicitly "opt in".

The opt-out strategy places less value on consumer information, assuming information is insensitive and can be shared, unless a consumer explicitly requests otherwise. The default with the opt-out strategy is to share information. If consumers do not want to share their information, they must explicitly "opt out".

2.2.6 Transparency

Transparency or consumer accessibility to collected data is another important consideration for companies. The trend is to allow consumers on-line access to their own data. Inevitably security issues are raised by opening up on-line access to data, since it may increase the risk of unauthorized access by third parties to an individual's personal information. However, if adequate authentication steps are in place, such as a user name and password, this mechanism can benefit both sides. Consumers benefit by being reassured as to the nature of the information maintained about them and the openness of their relationship with the business; additionally the company can minimize its costs by placing some of the responsibility of keeping information up-to-date on the consumer.

2.2.7 Information Use

The way in which collected information is used can be divided into three categories:

- Approved Intended uses. These are uses for which the company has notified the consumer and received approval. An example might be collecting and storing a consumer's billing information to streamline future purchasing.

- Non-Approved Intended uses. These are uses for which the company

has notified the consumer but not received approval or has not notified the consumer at all. An example would be selling a consumer's purchasing history to another company.

- Unintended uses. These are uses which neither the company nor the consumer anticipated or approved. An example would be a hacker gaining access to a database of credit card numbers and posting them on the Web.

Privacy legislation and technologies protect consumers by allowing them access to a company's list of non-approved intended uses so that educated choices can be made. It is also implied that protection against unintended uses must be provided (Madsen & Adams, 2002).

2.2.8 Information Sharing

At the moment, individuals will likely have separate collections of their personal information with multiple companies, resulting in duplication and administrative burden.

As these islands of consumer data are connected to each other, privacy will become even more of an issue. Although the power of such aggregation is obvious, from auto form-filling to optimized deliveries based on a user's agenda, the implications for privacy are also serious. Privacy of user information in this sharing model will require protected data storage, authentication and authorization of requesting applications and confidentiality of transmitted data (Tavani, 1999).

From the discussion in this section it should be clear that privacy is not an isolated issue, but that there are a number of issues and concepts central to implementing effective privacy protection. The next section will look at privacy legislation at both international and national levels.

2.3 Privacy Legislation

As has been established before, privacy is a fundamental human right. Privacy is recognized around the world and some countries include the right to privacy in their constitution (Constitution of the Republic of South Africa, 1996). "It is now common wisdom that the power, capacity and speed of

information technology is accelerating rapidly. The extent of privacy invasion - or certainly the potential to invade privacy - increases correspondingly" (EPIC and Privacy International, 2000c).

This section will examine both international and national privacy legislation that aims to deal with the new privacy issues that information technology has created.

2.3.1 Models Of Privacy Protection

According to EPIC and Privacy International (2000b) privacy legislation is implemented using four major models for privacy protection. The models are comprehensive laws, sectoral laws, self-regulation and technologies of privacy. These models can compliment or contradict each other depending on their application. Most countries use several of these models simultaneously. In the countries with the most efficient legislation all of the models work together to ensure privacy protection. The following sections will examine each model in more detail.

2.3.1.1 Comprehensive Laws

This is the preferred model for most countries adopting data protection laws. The model consists of a general law governing the collection, use and distribution of personal information, which is then enforced by an oversight body. The model was adopted by the European Union (EU) to ensure compliance with its data protection laws. A variation of this model was adopted in Canada and another variation is pending in Australia. Using this approach industry develops and enforces rules for privacy protection and is overseen by a privacy agency.

2.3.1.2 Sectoral Laws

Some countries, such as the United States, have avoided endorsing general rules in favor of specific sectoral data protection laws. Enforcement is usually achieved through a range of mechanisms. Problems with this approach are the lack of an oversight agency, and the fact that new legislation is required with the introduction of each new technology, often leading to protection lagging behind. Many countries use sectoral laws to compliment comprehen-

sive legislation by providing more detailed protection for certain categories of information.

2.3.1.3 Self-Regulation

Companies and industry bodies can achieve (in theory) data protection through various forms of self-regulation, in which they establish codes of practice and engage in self-policing. However, most efforts have been disappointing, with adequacy and lack of enforcement being the major problems. This model is currently promoted by the United States, Japan and Singapore.

2.3.1.4 Technologies Of Privacy

Due to the development of commercially available technology-based systems, individual users now also have the ability to implement privacy protection. Various technologies such as encryption, proxy servers and smart cards can be used to protect the privacy and security of communications. However, the secureness and trustworthiness of some of these systems, particularly those depending on the trustworthiness of third parties, remain questionable.

2.3.2 International Privacy Legislation

A number of international treaties and conventions, such as the International Covenant on Civil and Political Rights and the European Convention on Human Rights recognize the right to privacy. However, the broad statements of the right to privacy contained in these documents have been considered too vague to deal with the challenges presented by the development of information technology to privacy.

2.3.2.1 OECD

To deal with these challenges, the Organization for Economic Cooperation and Development (OECD) (1981) developed the "Guidelines for the Protection of Privacy and Transborder Flows of Personal Data" during 1981. In the same year the Council of Europe (COE) (1981) released the "Convention for the Protection of Individuals with regard to the Processing of Personal Data".

These two agreements have had a profound effect on the passing of legislation around the world, being widely used even outside the member countries.

Although the expression of data protection in various declarations and laws varies, all require that personal information must be (EPIC and Privacy International, 2000b):

- "obtained fairly and lawfully;
- used only for the original specified purpose;
- adequate, relevant and not excessive to purpose;
- accurate and up to date;
- accessible to the subject;
- kept secure; and
- destroyed after its purpose is completed".

2.3.2.2 European Directive

The most recent development in international data protection legislation has been the Council of the European Union's "Directive on the protection of individuals with regard to the processing of personal data and on the free movement of such data", also known as the European Directive (The Privacy Committee of New South Wales, 2002). The Directive aims to guarantee that the right to privacy is protected while ensuring that there are no unnecessary impediments to the flow of personal information within the EU. To achieve this aim a number of principles are set out to protect the "fundamental right to privacy" while still allowing a free market in information.

These principles set out standards for the collection, storage and use of personal information and are applicable to both public and private sector organizations. Strict requirements are also set out before personal information can be transferred to a country outside the EU. This includes an insurance of adequate levels of privacy protection in the destination country which will be assessed regarding all the circumstances surrounding the transfer. It is hoped that these restrictions on transborder flows of personal information will prove of great significance for future technological developments which

require an international exchange of information (The Privacy Committee of New South Wales, 2002).

2.3.2.3 The United States And Europe

Although the United States Privacy Act of 1974 protects records held by government agencies and requires the application of fair information practices, the right to privacy is not explicitly mentioned in the United States Constitution.

During 1998 Congress enacted the Children's Online Privacy Protection Act (COPPA), the first piece of legislation which specifically addresses children's online privacy. It requires commercial Internet sites to provide clear notice of their information gathering processes and obtain parental consent when obtaining personal information from children under 13. COPPA allows parents to access and check the information that has been collected and restrict its use. It also "prohibits conditioning a child's participation in a game, the offering of a prize or another activity, or the child disclosing more personal information than is necessary to participate in such activity" (Media Awareness Network, 1999).

Privacy laws in Europe are generally stronger than those in North America. While United States' and Canadian laws are primarily limited to the public sector, many European countries privacy laws control both the public and private sector. Although never formally ruled upon, there were serious doubts whether the sectoral and self-regulation approach to privacy protection employed in the United States would pass the adequacy test laid down by the European Directive (EPIC and Privacy International, 2000b).

In July 2000 a "Safe Harbor" agreement was formed between the EU and United States. The agreement rests on a self-regulatory system wherein companies simply promise not to breach their declared privacy practices. Limited enforcement and systematic review of compliance exist. Privacy advocates and consumer groups in both the United States and Europe are highly critical of the agreement, which they claim will provide European citizens insufficient personal data protection (EPIC and Privacy International, 2000b).

2.3.3 Privacy Legislation In South Africa

Privacy legislation in South Africa is still in its early stages of development. This is understandable when one considers that the first references to privacy has only been made in the Constitution of the Republic of South Africa (1996). Section 14 states that:

"Everyone has the right to privacy, which includes the right not to have -

(a) their person or home searched;

(b) their property searched;

(c) their possessions seized; or

(d) the privacy of their communications infringed".

Section 32 states that:

1. "Everyone has the right of access to -
   (a) any information held by the state, and;
   (b) any information that is held by another person and that is required for the exercise or protection of any rights;
2. National legislation must be enacted to give effect to this right, and may provide for reasonable measures to alleviate the administrative and financial burden on the state".

On January 21, 2000 the South African Parliament approved "The Promotion of Access to Information Act" (formerly named the Open Democracy Bill). The bill allows any person access to almost all information held by government bodies. It also allows individuals access, rights of correction and limitations on disclosure of information held by companies and other private bodies. Originally the proposed legislation also included comprehensive data protection provisions. However, the parliamentary committee requested that these provisions be removed and stated that "it would be dealing with the right to privacy in section 14 of the Constitution in an ad hoc and undesirable manner ... it is intended that South-Africa, in following the international trend, should enact separate privacy legislation. The Committee, therefore,

requests the Minister for Justice and Constitutional Development to introduce Privacy and Data Protection legislation, after thorough research on the matter, as soon as reasonably possible" (EPIC and Privacy International, 2000a). The Privacy and Data Protection Bill is still in its early stages of development.

Because South Africa does not have a privacy commission, the Human Rights Commission was given the mandate to implement the legislation. However, the commission has limited powers to enforce the Promotion of Access to Information Act.

The long awaited Green Paper on Electronic Commerce has been compiled by the Department of Communications in 2000 and raises a number of questions about new legislation (EPIC and Privacy International, 2000a):

- "Should South Africa adopt specific requirements for database owners and others collecting personal information, with regard to the treatment of such data?
- To what extent should companies be allowed/encouraged to adopt self-regulation standards for privacy protection?
- Should there be official minimum requirements for notice, choice, access and security practices concerning data collection and use?
- What penalties should be imposed for misuse of personal data, either by collecting information without consent, selling or distributing unauthorized data, or other abuses?
- If direct government regulation is to be considered, which bodies (e.g., the Human Rights Commission or a new agency) should be responsible for monitoring and enforcing privacy rules? What powers and limitations should such an agency have with regard to examining companies databases and practices?
- What role should other consumer protection bodies (e.g. the Consumer Council) play in this regard?"

In 2000, the South African Government began considering legislation that would replace the existing Interception and Monitoring Prohibition Act of

1992. The new Regulation of Interception (and Monitoring) of Communications Act will allow for the interception and monitoring of communications to protect national security, monitoring of users' Internet browsing and mobile phone location, and prohibits telecommunication services which do not have the capacity to be intercepted (Regulation of Interception (and Monitoring) of Communications Act, 2001). Both industry and civil rights groups have expressed strong criticism towards the proposal.

In 2002 the Electronic Communications and Transactions Act (2002) came into effect. This controversial document has been received with mixed reaction. Whereas some parts have been praised others have been severely criticized. With the focus on privacy, the Act contains several provisions regarding cryptography and authentication which "aim to balance the rights to privacy with national security and public interest considerations" (Fletcher, Gradidge, & Kingdon, 2002). This is intended to allow state authorities access to encrypted message content. According to Fletcher et al. (2002), the most disappointing part of the Act relates to the "protection of personal information and critical databases". The relatively lightweight data protection provisions leave South African consumers considerably more exposed than their counterparts in other jurisdictions. The impact of the Act on business as well as individuals remains to be seen.

Most recently the South African Law Commssion (2003) has initiated an enquiry to "investigate all aspects regarding the protection of the right to privacy of a person in relation to the processing (collection, storage, use and communication) of his, her or its personal information by the State or another person". In addition the enquiry will "recommend any legislative or other steps which should be taken in this regard". The result of the work done by the commission is expected to be formalized as the "Data Privacy Bill".

2.4 Conclusion

This chapter provided an overview of the concept of privacy. It defined the various aspects of privacy and gave an overview of privacy issues and concepts specifically related to the Internet. Finally, an overview of existing privacy legislation, both internationally and in South Africa, was given.

While privacy protection, or the lack thereof, may be the biggest concern of Internet users today it is not the only problem preventing the Internet from reaching its full potential. Drafting effective legislation that caters for new technologies has also proved a troublesome task.

A study by Consumers International (2001) revealed that existing privacy protection mechanisms in various countries are still not adequate. Many Internet sites in Europe and the United States still fall far short of international standards on data protection. In particular it was found that:

- "Just over two thirds of sites collect some sort of personal information and almost all of these sites asked for details that made it easy to identify and contact the person.
- The vast majority of sites gave users no choice about being on the site's own mailing list or having their name passed on to affiliates or third parties.
- Despite tight EU regulation, sites within the EU are no better at telling users how they use their data than sites based in the US. Indeed, some of the best privacy policies were found on US sites.
- The most popular US sites were more likely than the EU ones to give users a choice about being on the company's mailing list or having their name passed on, despite the existence of legislation which obliges EU-based sites to provide users with a choice.
- Only ten percent of sites targeting children asked children to get their parents' consent before giving personal information or to tell their parents afterwards".

The next chapter will look at privacy protection technologies that can help to implement existing legislation and provide Internet users with effective personal privacy protection.

Chapter 3

Privacy Protection Mechanisms

Not only individuals but also businesses have felt the effect of growing concerns regarding online privacy. According to Hinde (2001) "survey after survey has shown a growing reluctance to buy over the Web". The importance for businesses to implement effective privacy protection has grown greater as consumers increasingly resort to their own methods of privacy protection, such as supplying false personal details or opting out of transactions.

This chapter looks at the ways in which the computer industry has dealt with the privacy issues and legislation discussed in the previous chapter. Focus falls on the various mechanisms available to businesses and consumers to increase the level of individual privacy protection and thus gain back some of the trust that seems to have been lost.

3.1 Regulation Standards

Although there has been universal agreement on the importance of increasing the level of consumer privacy protection on the Internet, few globally effective solutions have emerged.

A report by the Electronic Privacy Information Center (EPIC) (1997) entitled "Surfer Beware: Personal Privacy and The Internet" ends with five recommendations for Web sites:

- "Web sites should make available a privacy policy that is easy to find. Ideally the policy should be accessible from the home page by looking for the word 'privacy'.

- Privacy policies should state clearly how and when personal information is collected.
- Web sites should make it possible for individuals to get access to their own data.
- Cookie-enabled transactions should be more transparent.
- Web sites should continue to support anonymous access for Internet users".

Apart from recommendations such as these, other efforts to improve privacy have centered around two main concepts. The first is the promotion of *self-regulation*. This is based on the premise that it is in the best interests of the online industry itself to provide a level of privacy that will win consumer confidence. The second has been the development of *"privacy protocols"*. The result of this development has been the Open Profiling Standard (OPS) (Hensley, Metral, Shardanand, Converse, & Myers, 1997), a proposed specification giving computer users greater control over how their personal information is exchanged on the Internet.

These concepts are examined next.

3.1.1 Self-Regulation

Through self-regulation it is hoped that companies and individuals will reach a satisfactory level of privacy without the need for government intervention. This approach has been promoted by the Department of Commerce and the Federal Trade Commission (FTC) in the United States.

Although self-regulation cannot be enforced, the FTC (Federal Trade Commision (FTC), 2000) has requested companies on the Internet to publish a privacy policy and to follow its "fair information practices" regarding consumer data. These practices include:

- Adoption and implementation of a privacy policy that takes into account consumer anxiety over sharing personal information online.
- Notice and disclosure of information collection and use practices.
- Choice and consent, giving users the opportunity to exercise control over their information.

- Data security and quality and access measures to help protect the security and accuracy of personally identifiable information.

Self-regulation puts the emphasis on consumer awareness, as it is the responsibility of Internet users to review a Web site's privacy policy and decide if they want to do business with that site. A privacy policy however does not guarantee compliance.

Companies in violation of their privacy policies can have sanctions brought against them and be forced to bring their practices in line with their privacy policies. Unfortunately there is no sanction for not providing a privacy policy at all. Privacy policies do not usually cover the actions of third parties, such as network advertisers.

Industry associations and third parties, also known as trust organizations, have developed seal programs to help guarantee consumers that companies are obeying pre-established standards and privacy policies. Trust organizations are discussed in more detail further on.

3.1.2 Open Profiling Standard

The OPS was born out of the desire to present Internet users with highly customized and personalized information, entertainment and services.

According to Hensley et al. (1997) the feasibility and widespread adoption of such products and services were being held back by two barriers:

- "The potential threat to individual privacy makes end users wary of sharing any information.
- Gathering the information that makes this personalization possible is inefficient".

The concern of Internet users about revealing personal information and the frustration they experience with having to provide much of the same information repeatedly over time outweighed the incentive of a personalized experience. To break down these barriers the OPS was proposed as a standard protocol for "exchanging profile information between individuals and service-providing parties, with built-in safeguards for individual privacy" (Hensley et al., 1997).

According to a press release by Firefly, Netscape, and Microsft (1997) the design of the OPS was guided by three principles that are intended to protect the interests of any party whose data is about to be exchanged. These principles are:

- Informed consent: before personal information can be collected the informed consent of the individual must be given. Complete knowledge of how the information is to be used must be given to the individual so that an informed choice regarding consent to its usage and exchange can be made.
- Value exchange: no person's information should be collected freely without offering the individual value in exchange.
- Control by source: access to information, as well as permission for dissemination, is exclusively controlled by its source.

OPS makes it possible for a Web browser to engage in a dialog with each visited Web site about the information the site requires and the information the individual has profiled to be released. Users store a personal profile of their information and can then authorize a Web site to access all, part or none thereof. With the consent of the user, additional information required by the site can be added to the profile (Magid, 1997).

Thus the OPS provides consumers with the security that no personal information can be divulged without their authorization, as well as saving them from repeated manual entry of information at multiple sites. Businesses are presented with the opportunity to create a personalized environment for their customers, thus realizing increased advertising and commerce revenue through more effective customer communications. OPS was incorporated into work being done at the World Wide Web Consortium (W3C) and has resulted in the Platform for Privacy Preferences (P3P) Project (Marchiori, 2002). This protocol is discussed in more detail in section 3.3 below.

3.2 Trust Organizations and Seals

To address online privacy concerns a number of organizations have developed Web seals. These seals are designed to let participants publicize that they

adhere to certain privacy policies and practices (Cavoukian & Crompton, 2000). By implementing sufficient privacy standards, enforcing compliance and resolving disputes, these organizations aim to promote privacy efforts around the world.

3.2.1 TRUSTe

TRUSTe (2003) is an international organization based on the notion of industry self-regulation. It is one of the oldest organizations of its kind and is widely respected amongst Internet users.

TRUSTe believes that Internet users have a right to online privacy and should have a choice as to how personal information is collected, used and shared by Web sites. By fostering an environment of openness and mutual trust, TRUSTe aims to protect privacy through open disclosure and giving individuals the power to make informed choices, thus helping to keep the Internet a free, comfortable and richly diverse community.

The cornerstone of the TRUSTe program is the "trustmark" (shown in Figure 3.1). This branded seal is awarded to Web sites that adhere to established privacy principles (as mentioned previously) and agree to ongoing TRUSTe oversight and consumer resolution procedures.

Figure 3.1: TRUSTe Trustmarks (http://www.truste.org)

When visiting a Web site that displays the TRUSTe seal, Internet users can be assured that the Web site will disclose (TRUSTe, 2003):

- "What personal information is being gathered about you.
- How the information will be used.
- Who the information will be shared with, if anyone.
- Choices available to you regarding how collected information is used.

- Safeguards in place to protect your information from loss, misuse, or alteration.
- How you can update or correct inaccuracies in your information".

To prevent fraudulent use of these seals each licensee must post a "click-to-verify" seal on their privacy statements. Clicking on this seal will take the user to a secure TRUSTe server which will verify that the site is a legal licensee of TRUSTe.

Although TRUSTe monitors licensees for compliance with their program, the public is still encouraged to stay vigilant and to report violations of privacy policies, misuse of the trustmark or privacy concerns pertaining to a specific Web site. This stresses, once again, the fact that the onus is on consumers to read the privacy policy and decide for themselves whether they want to disclose their personal information or not.

3.2.2 WebTrust

TRUSTe is not the only privacy assurance organization. Two other organizations that offer consumers the ability to identify sites that adhere to privacy standards by awarding them with a seal of approval are WebTrust (2003) and BBBOnline (2003).

The WebTrust seal, shown in Figure 3.2, is awarded to companies that consistently adhere to the standards established by the Canadian Institute of Chartered Accountants (CICA) and the American Institute of Chartered Public Accountants (AICPA).

Figure 3.2: WebTrust Seal (http://www.webtrust.net)

These globally recognized standards cover many areas such as privacy, security business practices/transaction integrity, availability, confidentiality and non-repudiation. Through independent and objective verification processes by CA's and CPA's, consumers are assured that the site they are visiting meets the high standards mentioned above and that it can be trusted with consumers' private information.

3.2.3 BBBOnline

A subsidiary of the Council of Better Business Bureaus, BBBOnline's "mission is to promote trust and confidence on the Internet through the BBBOnLine Reliability and Privacy Seal Programs" (BBBOnline, 2003). Once companies have been evaluated and confirmed to meet the programs' requirements they are allowed to display the seal, illustrated in Figure 3.3.

Figure 3.3: BBBOnline Seal (http://www.bbbonline.org)

Two seal programs are available, one regarding reliability and the other privacy. BBBOnline not only assures high standards of privacy but also facilitates a consumer dispute resolution system to resolve complaints.

3.2.4 Trust Seals: An Evaluation

These three leading online privacy seals were evaluated in a joint project by the privacy commissioners of Canada and Australia. The project, entitled "Web Seals: A Review of Online Privacy Programs" (Cavoukian & Crompton, 2000), tested each seal for the following key components:

- "sufficient privacy principles to which participating Web sites must adhere;

- a sound method for resolving disputes between consumers and Web sites; and
- a robust mechanism for ensuring that 'sealed' Web sites complied with the seal's standards".

Regarding privacy standards, out of a possible eight marks the scores awarded were: WebTrust 6.0, BBBOnline 6.25 and TRUSTe 6.375. For dispute resolution out of a total of six points the scores were: WebTrust 4.58, TRUSTe 4.65 and BBBOnline 5.05.

Although none of the seals completely satisfied the criteria, each had its own strengths. WebTrust offered the most rigorous compliance system. BBBOnline offered the most customer-friendly dispute resolution system. In terms of privacy principles TRUSTe scored the highest. The project concluded that "seals are playing a valuable educational role in promoting privacy awareness in the minds of both consumers and businesses alike. This educational role is, in our view, both positive and beneficial" (Cavoukian & Crompton, 2000).

3.3 Platform for Privacy Preferences

In 2002 the World Wide Web Consortium released P3P as a recommendation after almost five years of development. In short P3P "is a protocol designed to inform Web users of the data-collection practices of Web sites"(Marchiori, 2002). So far it has been the biggest effort towards creating a technical mechanism for protecting the privacy of individuals and is the most visible industry-supported privacy protection tool.

3.3.1 P3P Model

P3P is based on a model of privacy developed by the US Department of Commerces National Telecommunications and Information Administration (NTIA) and the FTC (Hochheiser, 2002). The model comprises five aspects which are summarized in Table 3.1 and considered in more detail.

Notice/Awareness requires companies to let consumers know that information is being collected about them, as well as what the information is, how

it will be used and protected, and whether it will be shared with any other parties.

Choice/Consent aims to give consumers the option of opting-in or opting-out of these information gathering practices. This decision should be based on the notice given in the privacy policies.

Access/Participation allows consumers to examine, correct or request deletion of their records after this information has been gathered.

Security/Integrity mechanisms should be in place to guard against unauthorized access of consumer information. Both physical and logical safeguards should be in place to comfort consumer concerns.

Enforcement/Redress requires that companies follow their privacy policies and that there are mechanisms to cope with privacy abuse. Trust organizations play a major role in guaranteeing consumers that companies are obeying their policies and standards.

P3P was clearly developed to address notice and choice and provides only limited support for the remaining privacy principles listed above. While access is controlled, P3P does not address security or enforcement issues. As such its developers have acknowledged that it is not a complete solution to all of the concerns regarding Internet privacy (Grimm & Rossnagel, 2000).

The notions of notice and choice imply that users should be made aware of the information collection practices of an entity before any information is collected and that options should be given as to the manner in which personal data is used. Thus it is hoped that an atmosphere of greater trust between Web sites and users can be established.

Any Web site wishing to gather information about users browsing their site should give *notice* on what data will be collected, who the recipients of the data will be and how they intend to use the information. A user browsing a site with a P3P policy then has access to the site's privacy policies and has the *choice* of accepting cookies from the site or not using the site at

Table 3.1: Privacy Model

Notice/Awareness	Notice of information practices
Choice/Consent	Options to the use of collected data
Access/Participation	Access to collected data
Integrity/Security	Accuracy and protection of data
Enforcement/Redress	Ensure application of principles

all (Bathe, 2002).

When a P3P enabled browser visits a Web site the P3P policy for that page is automatically fetched. This policy is then compared with the user's preferences and the appropriate action taken. If the data collection practices are acceptable according to user preferences the page is displayed normally. However, if a clash occurs the user is given a warning message explaining the contents of the P3P policy and given the choice to either continue or cancel the current action (Reagle & Cranor, 1999).

P3P does not provide any mechanisms for limiting the collection of data and the use thereof. Rather these practices are assumed to occur indirectly as the result of consumer-friendly privacy policies. The privacy of e-mail and other online activities are also beyond the scope of the project. Most importantly P3P does not provide any mechanism to ensure that Web sites act according to their policies. Instead it is "complementary to laws and self-regulation programs that can provide enforcement mechanisms" (Cranor, 2002).

3.3.2 P3P Critique

P3P critics have raised a number of important questions as to the effectiveness of the protocol. Important issues that have arisen include the following:

- Many people have a view of privacy that is different from the notice/choice model underlying P3P. To them P3P fails to address privacy and the only effective control would be standards and new legislation (Hochheiser, 2002).

- No means exist to enforce the promises made to users in privacy policies. Companies know that there is no enforcement and very little incentive exists for companies to adhere to their policies (Coyle, 1999).

- Web sites that lack P3P code will be excluded even though their privacy practices may exceed P3P compliant sites (Electronic Privacy Information Center (EPIC), 2000).

If these issues cannot be addressed successfully P3P is more "likely to undermine public confidence in Internet privacy" (Electronic Privacy Infor-

mation Center (EPIC), 2000) than to provide an atmosphere of trust between users and Web sites.

3.4 Other Privacy Tools

"An interesting consequence of the rising consumer concern over loss of privacy ... is the rise of Internet firms that aim to help consumers preserve their anonymity in the new environment of personalized marketing" (Robertson & Sarathy, 2002). The word "privacy" has become a major selling tool and companies have not hesitated to use it. Almost every program remotely related to privacy proclaims its ability to protect the information of users while surfing the Internet.

Many programs claim to enhance user privacy by controlling the use of cookies, blocking pop-up advertisements and preventing malicious scripts from running. These features are so common that they are built into many Anti-virus and Firewall software packages as well as countless stand-alone applications.

This the ability to clean your URL and browser history, as well as any temporary Internet files stored on your computer, is also considered as privacy enhancing and many programs are available for these purposes.

The Electronic Privacy Information Center keeps a comprehensive list of privacy-enhancing tools. According to them the techniques that protect privacy are those that "minimize or eliminate the collection of personally identifiable information" (Electronic Privacy Information Center (EPIC), 2000). These tools allow users to become anonymous, denying marketers the ability to collect information about them. The tools they list provide users with services such as anonymous surfing, cookie protection, HTML filtering, email and file privacy and encryption[1].

By making use of these technologies users can genuinely enhance their privacy on the Internet.

[1]The complete list is available at `http://www.epic.org/privacy/tools.html`.

3.5 Cookies

One of the most controversial topics related to privacy concerns is the use of HTTP cookies (Kristol & Montulli, 2000). Cookies, RFC 2965, were developed to provide Web sites with a mechanism "to maintain state in the otherwise stateless HTTP protocol" (Kristol, 2001). Essentially a cookie is a file passed between client and server which allows the server to identify returning clients. As such, the idea and use of cookies seem very useful and innocent.

The problem lies in the fact that there are some controversial uses of cookies which have become very common.

One such controversial use relates to the issue of personalization. Marketers argue that by collecting information about consumers they can sell more effectively and efficiently. Advertisements can be personalized increasing the chances of users responding positively.

Cookies allow this personalization to take place by allowing Web sites to identify users and their browsing patterns. When visiting a Web site a third party, such as an advertising company, can set a cookie on your computer even though the visited page is not on their server. Whereas users can expect a cookie from the Web site they are visiting they have no reason to expect, or know about, cookies from third parties.

These cookies eventually allow a history of browsing behavior to be accumulated and linked to other information about a user. Even though such information helps to provide Web sites with a valuable source of revenue, consumers still want to protect their privacy and have expressed anger at and oppose such business models (Robertson & Sarathy, 2002).

"These so-called third-party cookies practices are clearly privacy-invasive" (Electronic Privacy Information Center (EPIC), 2000). To combat this invasion, projects such as P3P aim to give users greater control over cookies by providing more information for deciding which cookies to accept or block. In this way at least a measure of control is provided over the information that is gathered about users without their knowledge.

3.6 Conclusion

In a report to the US Congress the FTC has given evidence "that the self-regulatory approach towards protecting online privacy had failed" (Hinde, 2002). The lack of self-regulatory mechanisms being adopted by industry has led to the belief that legislation is needed to protect online consumer privacy. Nevertheless, debate still continues as to the benefits of each approach.

What has been agreed upon is the need to effectively enforce existing laws and educate both consumers and companies. According to Ann Cavuokian of the Ontario Information and Privacy Commissioner "privacy is the next business imperative. The fact is that good privacy is good business – it fosters trust, builds consumer confidence, strengthens brand recognition, increases customer loyalty and ultimately, delivers competitive advantage" (Hinde, 2002). As such the status of privacy has been raised from philosophy to business necessity (Hinde, 2001). If businesses can win the trust of consumers this will certainly be a win-win situation for both sides.

One feels that significant steps to improving online privacy will only be made once consumers have learnt and implemented proper privacy practices, and organizations adopted the necessary mechanisms and regulations to gain consumer trust.

Chapter 4

Web Services

As mentioned in the introduction, the focus of this study is the implications of effective privacy support in service-based environments. This chapter will examine service-based environments with specific focus on Web Services.

Backed by major companies like IBM, Microsoft and Sun, it is no wonder that Web Services is significantly influencing the evolution of the Internet. Sceptics of this technology are increasingly harder to find as enthusiasm grows over the possibilities that Web Services hold. Businesses, online consumers as well as developers are set to gain from the technology.

This chapter will provide a detailed picture of Web Services, examining the drive behind it and its essential supporting technologies.

4.1 Background

One of the biggest problems faced on the Internet is the integration of disparate systems. As each vendor releases proprietary interface protocols the difficulty of integrating different systems increases.

Web Services aim to provide a new standardized way of application integrating and interoperability to solve this problem. "Web Services opens up a world where we are able to share and request for information with those we want with ease and speed like we never did before..." (Peng, 2002).

By definition Web Services is "loosely coupled software components delivered over Internet-standard technologies that can automatically maintain information based on open standards that span multiple platforms" (Hyman, 2002). Effectively a Web Service is a business service that can be accessed

by applications over public or private networks using standard protocols.

By utilizing open standards Web Services is set to streamline business integration between suppliers, partners and customers. The benefits to be gained are significant, as companies will be able to adapt to market changes, introduce new applications and share information quicker and easier. Web Services is designed to free companies from huge investments by giving them a cheaper, faster and easier alternative to application integration (Peng, 2002).

4.2 Supporting Technologies

To implement these goals of integration and interoperability Web Services relies on various supporting technologies. Very importantly, the Web Services standards and infrastructure are designed and developed to be extensible, enabling new standards and technologies to be used as they emerge (Newcomer, 2003, p. 14).

The core issues that need to be addressed to facilitate remote communication between two applications are discovery, description, message format, encoding and transport (Short, 2002, p. 4). In a Web Service environment each of these rely on specific technologies, illustrated in Figure 4.1.

Discovery allows a client application to resolve the location of a Web Service which it needs access to. Universal Description, Discovery, and In-

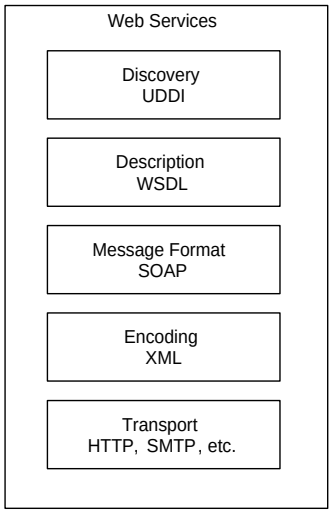

Figure 4.1: Web Service Technologies

tegration (UDDI) (OASIS, 2003) provides a common way to advertise Web Services. UDDI is a centralized directory which allows clients to search for Web Services using a host of criteria.

Description enables a client to interact properly with a Web Service once it has been discovered. Information such as the transport protocol to be used, the address of the service and the response messages to expect must be ascertained before a service can be used. This information is provided by a Web Services Description Language (WSDL) (Christensen, Curebera, Meredith, & Weerawarana, 2001) document which fully describes a particular Web Service.

When exchanging data the client and Web Service need to agree on a common *message format*. This ensures that both parties will interpret the encoded data properly. The Simple Object Access Protocol (SOAP) (Gudgin, Hadley, Mendelsohn, Moreau, & Nielsen, 2003a) was chosen to provide a standard way of formatting messages. Using SOAP every message can contain additional metadata along with the body of the message.

Each transmission between client and Web Service needs to *encode* the data into the body of the message. The Extensible Markup Language (XML) (Bray, Paoli, Sperberg-McQueen, & Maler, 1999) is the natural choice as it is an industry standard that provides cross-platform support, is lightweight and has excellent tool support.

Once the message has been properly formatted it must be *transported* using some protocol. The Hypertext Transfer Protocol (HTTP) (Fielding et al., 1999), and Simple Mail Transfer Protocol (SMTP) (Klensin, 2001), are both proven Internet protocols. As the infrastructure to support these protocols are universally available they are the ideal transfer mechanism.

A discussion of the core standards of the aforementioned technologies follow.

4.3 XML

While Hypertext Markup Language (HTML) (Raggett, Le Hors, & Jacobs, 1999) documents are portable and easy to use they suffer from a few very important drawbacks. These include the lack of any real document structure, the absence of semantic information, the difficulty in reusing information and

the fact that HTML is a rigid standard that cannot be extended.

XML aims to address these issues while maintaining the advantages of HTML. The following is an example of an XML document:

```
<?xml version="1.0"?>
<products>
    <product id="1">
        <name>XML Book</name>
        <price currency="ZAR">300.00</price>
    </product>
</products>
```

As no tags are predefined in XML the author is free to create the needed tags. The above document could represent a simple product catalog. For each item a *product* element is defined with an *id* attribute. This element in turn contains a *name* element describing the product and a *price* element with a *currency* attribute. Each document is structured and contains semantic information. With XML the focus is on data-oriented documents and not on presentation detail.

A number of standards that complement XML have also been developed. Two of these companion standards that are very important in relation to Web Services are XML Namespaces (Bray, Hollander, & Layman, 1999) and XML Schemas (Fallside, 2003). These standards are examined next.

4.4 XML Namespaces

The purpose of namespaces is to lift the ambiguity of XML tags by qualifying them. Namespaces in XML has been endorsed as a W3C Recommendation and even though it is a fairly short document, about 10 pages, it is still a source of much confusion in XML, especially for newcomers to the technology (Skonnard, 2001).

4.4.1 What Is A Namespace?

A namespace is simply a collection of names in which all names are unique. Examples of namespaces are the names of tables in a relational database, programming type identifiers or Internet domains. “Any logically related set of names in which each name must be unique is a namespace” (Skonnard, 2001).

The W3C give the following definition of namespaces in XML: “An XML namespace is a collection of names, identified by a URI reference, which are used in XML documents as element types and attribute names” (Bray et al., 1999).

Namespaces make it easier to come up with unique names by restricting the uniqueness to a more limited context. To maintain the integrity of a namespace, a namespace authority must ensure that new additions to a namespace do not already exist in that namespace. If duplication of names occur the namespace would lose its uniqueness and would no longer be officially considered a namespace.

In order to be useful namespaces themselves must be given names. Once a namespace has been named it is possible to refer to its members. In Figure 4.2 two namespaces are represented. Although these namespaces contain similar members it is possible to uniquely identify them through namespace-qualified names.

If the names of namespaces cannot be guaranteed to be unique it is necessary to place them in a namespace of their own. If, in the above example, there are multiple hardware stores, placing them each in a distinct namespace would resolve the conflict. This pattern can be repeated as many times as necessary to ensure the uniqueness of a name.

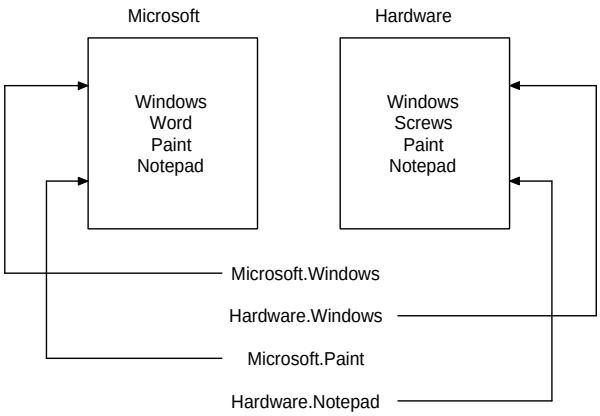

Figure 4.2: Unique Namespaces

4.4.2 Naming Namespaces

As defined by the W3C, XML namespaces must conform to the syntax for Uniform Resource Identifier (URI) references. URI references are strings of characters which identify abstract or physical resources.

Figure 4.3 illustrates the two types of URI references that may be used as a namespace identifier. These are Uniform Resource Locators (URL) (Berners-Lee, Masinter, & McCahill, 1994), RFC 1738, or Uniform Resource Names (URN) (Moats, 1997), RFC 2141.

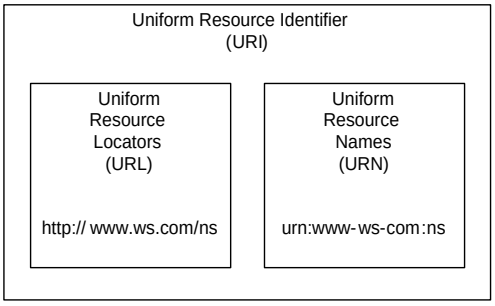

Figure 4.3: URI Types

XML processors treat namespace identifiers as opaque strings and never as resolvable resources. "URI references which identify namespaces are considered identical when they are exactly the same character-for-character" (Bray et al., 1999). Thus Figure 4.3 shows two different namespaces.

4.4.3 Using Namespaces

Using a namespace can be seen as the process of using one or more elements or attributes from a given namespace in an XML document.

Because URI references can contain characters not allowed in names they cannot be used directly. A prefix, which is mapped to a URI reference through a namespace declaration, selects a namespace. The prefix is considered in-scope on the declaration element as well as any descendant elements. For example:

```
<?xml version="1.0"?>
<bk:book xmlns:bk="http://www.book.com/ex">
    ...
</bk:book>
```

On line 2 the *bk* prefix is bound to `http://www.book.com/ex`. This prefix applies to the *book* element as well as its contents.

XML element and attribute names are made up of two parts: a namespace prefix and a local name, separated by a single colon. This two part name is known as the qualified name. For example:

```
1 <?xml version="1.0"?>
2 <bk:book xmlns:bk="http://www.book.com/ex">
3     <bk:value bk:currency="$">9.99</bk:value>
4 </bk:book>
```

Line 3 defines a *value* element with a *currency* attribute. Both belong to the namespace `http://www.book.com/ex` declared on line 2. The qualified name for the *value* element would be *bk:value*.

Namespace declarations apply to the element in which it is specified as well as all elements within the context of that element. Elements and attributes from more than one namespace can be used by having multiple namespace declarations. Namespaces can also be overridden by redeclaring the prefix. For example:

```
1 <?xml version="1.0"?>
2 <bk:book xmlns:bk="http://www.book.com/ex"
3          xmlns:isbn="http://www.isbn.com/ex">
4     <bk:title>Namespaces</bk:title>
5     <isbn:number>1234567890</isbn:number>
6     <bk:author xmlns:bk="http://www.author.com/ex">
7         John Doe
8     </bk:author>
9 </bk:book>
```

Two namespaces are declared in the *book* element on line 2. The *isbn* prefix is used on line 5 in the *number* element. The *author* element on line 6 redeclares the *bk* prefix and binds it to `http://www.author.com/ex`.

Any element without a prefix is considered to be from no namespace, except when a default namespace is declared. A default namespace applies to the element in which it is declared as well as all elements within the context of that element. A default namespace can be set to an empty string, having the same effect as there being no namespace. Default namespaces do not apply to attributes. For example:

```
1 <?xml version="1.0"?>
2 <book xmlns="http://www.book.com/ex">
```

```
3     <title>Default Namespace</title>
4     <author xmlns="">John Doe</author>
5 </book>
```

The *book* and *title* elements on lines 2 and 3 belong to the default namespace `http://www.book.com/ex` declared on line 2. The *author* element on line 4 does not belong to a namespace.

4.4.4 Summary

Namespaces are collections of names in which all names are unique. In XML they provide a simple method for qualifying element and attribute names by associating them with namespaces identified by URI references. This makes it possible to render element and attribute names less ambiguous by giving them unique identifiers. More information on namespaces can be obtained at `http://www.rpbourret.com/xml/NamespacesFAQ.htm`.

4.5 XML Schemas

Schemas provide a standard way of describing the structure and the type of information that should be contained within an XML document (Short, 2002, p. 59). Schemas have gradually replaced Document Type Definitions (DTD) originally introduced with XML. DTD's are considered inferior because they cannot describe datatypes and are not XML based.

The specification is divided into three parts:

1. Part 0: Primer (Fallside, 2003) provides developers with an explanation of what schemas are and how to use them.

2. Part 1: Structures (Thompson, Beech, Maloney, & Mendelsohn, 2001) defines constructs for describing the structure of XML documents.

3. Finally, XML Schema Part 2: Datatypes (Biron & Malhotra, 2003) defines a set of simple datatypes.

4.5.1 Schema Concepts

Schemas can be used to control which elements can be used in a particular context. Element content can be controlled as well as attributes and their

values. These features can be useful in various scenarios. Authors can be forced to follow a specific document model. Documents can be validated before transforming and exchanging them with a business partner thus reducing errors. Large collections of documents can be more easily maintained by making sure that each document follows the same model.

One of the most valuable features for Web Services is the fact that XML Schema defines a core set of datatypes that are completely platform independent. Along with these built-in types XML Schema also allows custom simple types to be defined as subsets of the predefined types. In this way values can be restricted to more specific ranges. Different simple types can also be arranged into a structure, known as a complex type. These features provide XML with a powerful and extensible type system.

An XML document that conforms to an XML Schema is often referred to as an instance document. Below a simple XML Schema document is shown:

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
            targetNamespace="http://example.org/"
            xmlns="http://example.org">
<xsd:element name="publications">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="book" maxOccurs="unbounded">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="title"
                            type="xsd:string"/>
                        <xsd:element name="author"
                            type="xsd:string"/>
                        <xsd:element name="description"
                            type="xsd:string"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
</xsd:schema>
```

An XML document conforming to this schema is shown next:

```
<?xml version="1.0"?>
<publications>
    <book>
        <title>Mastering XHTML</title>
```

```
5         <author>Ed Tittel</author>
6         <description>
7             Newly revised and updated, Mastering XHTML
8             is a complete guide to the markup language
9             that is leading the world of Web development
10            from HTML to XML.
11        </description>
12    </book>
13    <book>
14        <title>
15            Java Developers Guide to E-Commerce
16            with XML and JSP
17        </title>
18        <author>William Brogden</author>
19        <description>
20            Covering the latest Servlet and JSP APIs
21            and the current XML standard, this book
22            guides you through all the steps required
23            to build and implement a cohesive, dynamic,
24            and profitable site.
25        </description>
26    </book>
27 </publications>
```

The schema constrains the XML document to a single *publications* element[1] which can contain any number of *book* elements. Each of these has a *title*, *author* and *description* element with contents of type string which must occur in that sequence.

4.5.2 Summary

XML Schemas provide a comprehensive and extensible way of describing the structure and datatypes that should appear within an instance document. These features are used extensively throughout the rest of the Web Services architecture. For a complete examination of schemas see the book "XML Schemas" by Valentine, Dykes, and Tittel (2001).

4.6 SOAP

SOAP provides Web Services with a standard way of packaging messages. One of the primary reasons why it has received attention is widespread in-

[1] By definition an XML document can only have one root element.

dustry support. It is the first protocol of its kind to be accepted by almost all major software companies in the world (Short, 2002, p. 29).

In 2003 the W3C released SOAP 1.2 as a Recommendation. The specification consists of four parts, the most important dealing with the Messaging Framework (Gudgin, Hadley, Mendelsohn, Moreau, & Nielsen, 2003b) and Adjuncts (Gudgin, Hadley, Mendelsohn, Moreau, & Nielsen, 2003c).

As a lightweight messaging protocol SOAP offers many advantages. It is not tightly coupled to any programming languages or transport protocols, allowing more flexibility in development and implementation. It leverages existing standards such as XML and HTTP. It enables interoperability across multiple environments allowing applications on different platforms to communicate via SOAP messages.

4.6.1 Message Structure

Every SOAP message consists of an envelope that contains the body of the message and any header information for describing the message. The basic structure is illustrated in Figure 4.4.

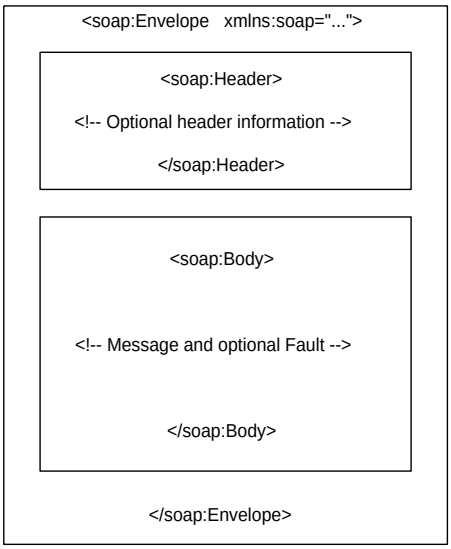

Figure 4.4: SOAP Message

The *envelope* element is always the root element of a SOAP message. This makes it easy for applications to identify SOAP messages by looking at the name of the root element. The *envelope* contains an optional *header* element

followed by a mandatory *body* element.

The *header* element carries additional information about the message that might not be appropriate to encode in the body. This could include security information such as authentication or digital signatures, routing information, payment details, compression data, etc. Because headers are optional and the recipient can choose to ignore them, SOAP specifies a *mustUnderstand* attribute to distinguish critical information that must be processed. For example:

```
1  <?xml version="1.0"?>
2  <soap:Envelope
3  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
4  <soap:Header>
5      <credentials soap:mustUnderstand="1">
6          <username>John</username>
7          <password>Doe</password>
8      <credentials>
9  </soap:Header>
10 ...
```

Lines 5–8 define a set of security credentials. The *mustUnderstand* attribute of the *credentials* element on line 5 specifies that these headers must be processed. If the receiver is not able to process such a header the message should not be processed and a fault should be returned.

The *body* element represents the message payload. It can contain any number of elements from any namespace. The body can also contain a *fault* element for representing errors. If an error occurs while processing a SOAP message the *fault* element can provide details about the error which may help clients to diagnose the problem. For example:

```
1  <?xml version="1.0"?>
2  <soap:Envelope
3  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
4  <soap:Body>
5      <soap:Fault>
6          <soap:faultcode>Client.Security</soap:faultcode>
7          <soap:faultstring>Access denied</soap:faultstring>
8      </soap:Fault>
9  </soap:Body>
10 </soap:Envelope>
```

Line 5 defines a *fault* element whose contents, the *faultcode* and *faultstring* elements, contain information on the nature of the error that occurred.

Default fault codes are provided to describe errors related to version mismatches, failure to process an element containing the *mustUnderstand* attribute, incorrect message content and server errors.

4.6.2 Message Styles

SOAP messages can be classified into two fundamental categories: document and procedure style messages. Procedure messages provide two-way communication and are commonly referred to as Remote Procedure Call (RPC) (Srinivasan, 1995), RFC 1831, messages.

Document messages indicate that the body simply contains an XML document whose format is agreed upon by both sender and receiver. This generally facilitates a one-way communication. An example might be a purchase order.

"One of the original design goals of SOAP was to provide an open and standard way to facilitate RPCs using Internet technologies such as XML and HTTP" (Short, 2002, p. 38). RPC messages contain an XML representation of a method call and provide the ability to invoke a method and to return a response. Most discussions of SOAP have focussed on its RPC abilities.

To implement the request/response behavior needed by RPC two messages are needed. Consider a simple C# function which adds two numbers:

```
public int Add(int x, int y)
{
    return x + y;
}
```

The request SOAP message would be encoded as follows:

```
1  <?xml version="1.0"?>
2  <soap:Envelope
3  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
4  <soap:Body>
5      <Add>
6          <x>2</x>
7          <y>3</y>
8      </Add>
9  </soap:Body>
10 </soap:Envelope>
```

The name of the function is contained within the body on line 5 as an *add* element with each input parameter represented as a subelement within (lines 6 and 7). The order and types of the parameters must also match.

After receiving and processing the message the response message generated by the remote application would look like this:

```
1 <?xml version="1.0"?>
2 <soap:Envelope
3 xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
4 <soap:Body>
5     <AddResult>
6         <result>5</result>
7     </AddResult>
8 </soap:Body>
9 </soap:Envelope>
```

The result of the method returns the data as a subelement within the body (line 5) with the name of the method and *Result* appended to it. On line 6 the *result* element contains the return parameter of the method call.

This naming convention is not dictated by the specification but is accepted as the standard method of encoding RPC style SOAP messages.

4.6.3 Protocol Binding

Even though SOAP is not tied to a particular transport protocol the majority of messages are sent using HTTP. However, any protocol capable of carrying text can be used.

Using HTTP offers many advantages such as:

- its robust supporting infrastructure,
- the fact that it is stateless,
- ensuring reliable communication across the Internet,
- its simplicity and network support, and
- the fact that it is firewall friendly.

How a message is carried by a particular protocol is known as the protocol binding. The SOAP specification only describes one protocol binding, sending SOAP messages via HTTP POST. An example request message might be:

```
POST /path/webservice.asmx HTTP/1.1
Content-Type: text/xml
```

```
3 SOAPAction: "" Content-Length: 200
4
5 <?xml version="1.0"?>
6 <soap:Envelope ...
```

The *Content-Type* header on line 2 must be set to text/xml. An additional *SOAPAction* header (line 3) is also defined which represents the intent of the message. It can consist of a value or be left blank if the intent is contained in the HTTP request header.

4.6.4 Summary

SOAP provides the underlying messaging protocol of Web Services. This protocol is both simple and extensible and can be used over multiple transport protocols with a variety of programming models (Skonnard, 2003). More information on SOAP can be found in the book "Advanced SOAP for Web Development" by Livingston (2002).

4.7 WSDL

The Web Services Description Language is a specification that defines how Web Services can be described using XML. WSDL "makes it easy to reap the benefits of SOAP by providing a way for Web service providers and users of such services to work together easily" (Tapang, 2001). A WSDL document provides an interface description for a Web Service, allowing automatic integration of the service in a language- and platform-independent way.

WSDL describes a Web Service using several major pieces of information which include:

- Interface information describing the operations performed by the Web Service.
- The data types for the messages used by the Web Service.
- Binding information about the communication protocols to be used.
- Information for locating the specified service.

The specification itself can be divided into six major elements, illustrated in Figure 4.5, which are discussed below.

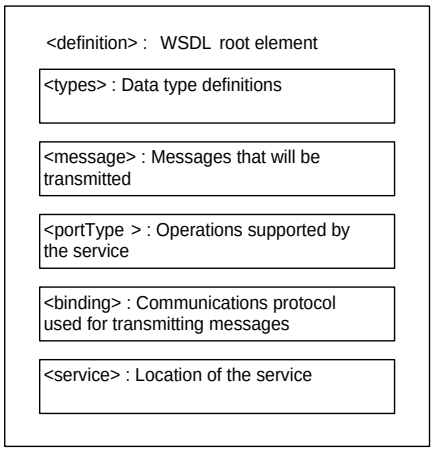

Figure 4.5: WSDL Specification Summary

4.7.1 Definitions Element

The *definitions* element must be the root element of all WSDL documents. It defines the name of the Web Service, declares multiple namespaces used throughout the remainder of the document, and contains all the other elements.

4.7.2 Types Element

The *types* element defines all the data types that are used by the Web Service. For maximum platform neutrality the XML Schema syntax is used by default. It may be omitted if no types need to be declared.

4.7.3 Message Element

Each *message* element describes a request or response operation. It defines the name of the message and can contain multiple *part* child elements, which refer to message parameters or return values. Each *part* element has a name and type attribute, similar to function parameters.

4.7.4 PortType Element

The *portType* element defines the Web Service operations that can be performed and the messages that are involved. Multiple messages can be combined into a single operation. WSDL defines four operation types:

- One-way – The service receives a message without responding.
- Request-response – The service receives a message and then sends a response.
- Solicit-response – The service sends a message and receives a response.
- Notification – The service sends a message, not waiting for a response.

When receiving and sending messages an operation defines an input and output element respectively whose attributes refer to the relevant *message* elements. The operation types are illustrated in Figure 4.6.

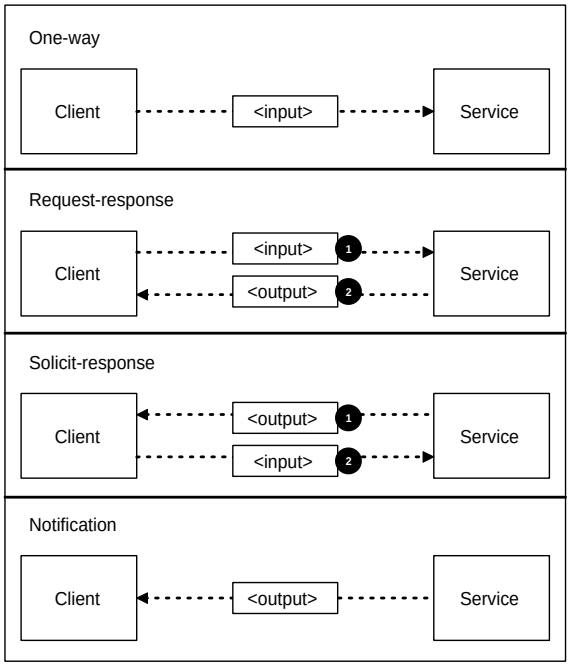

Figure 4.6: WSDL Operations

4.7.5 Binding Element

The *binding* element defines how a *portType* operation will be transmitted over the wire. When binding to SOAP it enables one to specify details such as SOAP headers, encoding style and the SOAPAction HTTP header.

4.7.6 Service Element

The *service* element defines the address for invoking the specified service and usually includes a URL for invoking the SOAP service.

4.7.7 Summary

WSDL represents a fundamental part of the Web Service architecture. By providing a common language for describing services and a platform for automatically integrating those services, it enables applications to invoke services easily. More information on WSDL can be obtained at `http://www.w3schools.com/wsdl`, as well as the book "Web Services Essentials" by Cerami (2002).

4.8 UDDI

Universal Description, Discovery, and Integration provides a "standardized method for publishing and discovering information about web services" (Chappell & Jewell, 2002, p. 98). It is an initiative that has received substantial support from a number of technology companies, such as Microsoft and IBM[2].

UDDI attempts to create a global, platform-independent, open framework to allow companies to describe the services they offer, discover other businesses and integrate business services. In more general terms "UDDI provides a central directory service for publishing technical information about Web services" (Short, 2002, p. 256).

4.8.1 UDDI Architecture

The UDDI infrastructure is composed of registries and registrars. Each registry contains a complete copy of the UDDI directory, while a registrar provides registration services to customers.

Each registry (or node) has its data synchronized through replication. Thus accessing any individual node provides the same information and quality of service as any other node. When information is added to the UDDI directory it is done at a single node which becomes the master owner of that

[2] A full list of participants is available from `http://www.uddi.org/community.html`.

content. Any subsequent updates or deletes of the information must occur at the node where the information was inserted.

Figure 4.7 shows a user updating information in the directory through a registrar and another user accessing the information.

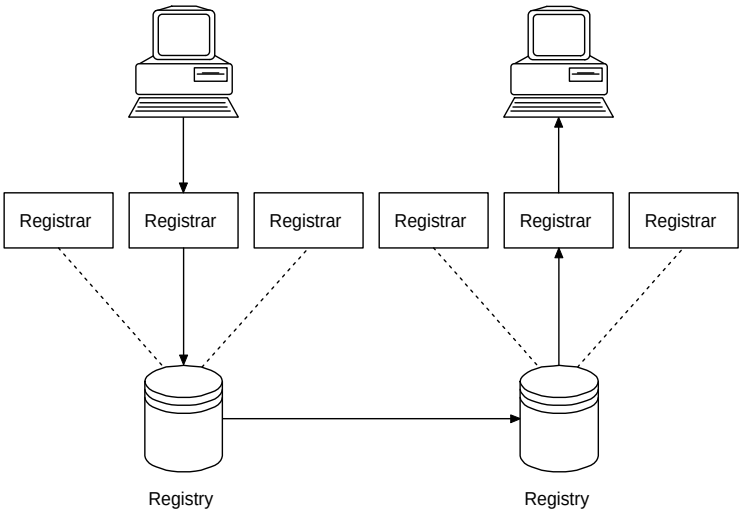

Figure 4.7: UDDI Architecture

Companies can also create private nodes that are not part of the UDDI directory. They can then group these nodes together, replicating information between them, without having any interaction with the "public" UDDI directory.

4.8.2 UDDI Data Structures

Each registry is in itself a Web Service and exposes a SOAP-based Application Programming Interface (API) for accessing and manipulating entries in the directory. In addition to the methods for interacting with the data, several data structures are also defined. Figure 4.8 shows the relationships between the primary data structures.

A *businessEntity* defines the business itself. The structure represents basic information about the business such as contact information, categorization, descriptions and relationships to other businesses. Each company must establish a unique *businessEntity*.

A *businessEntity* contains one or more *businessService* structures. A *businessService* describes a collection of services provided by the business

and represents a single, logical service classification. These services can be Web Services or non-electronic services. A *businessService* element can also be reused by several *businessEntity* elements.

A *businessService* contains one or more *bindingTemplate* structures. A *bindingTemplate* specifies the location of technical descriptions and the access point URL of a service. An optional text description of the Web Service may also be present. It does not contain the details of the service's specifications. It also contains a reference to one or more *tModel* structures.

A *tModel* defines a particular specification for a service. It is used for determining the specifics of how to interact with a particular Web Service. Companies can used the *tModel* information to determine whether a Web Service satisfies their business requirements.

4.8.3 Summary

"Universal Description, Discovery and Integration (UDDI) is a directory where businesses can register and search for Web services" (Shohoud, 2003, p. 409). UDDI promises to solve the current problems businesses have in reaching their customers and partners with information about their products and services. By describing services and business processes programmatically it can help companies integrate into each other's systems and processes.

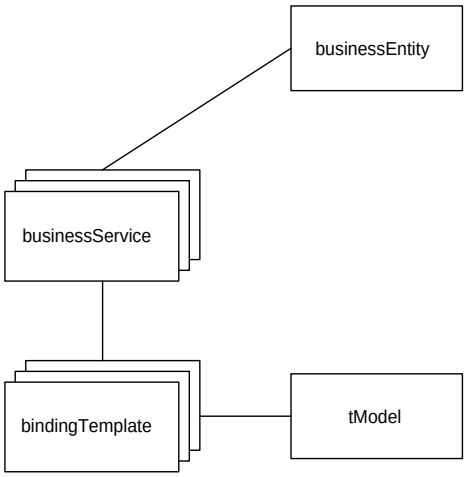

Figure 4.8: UDDI Data Structure Relationships

4.9 Conclusion

Web Services is an umbrella term describing a collection of standard protocols and services used to achieve a base-line level of integration and interoperability between applications.

Although these technologies and the ways in which developers leverage them are still in their infancy, the industry support that Web Services have received is phenomenal. "Never before have so many leading technology companies stepped up to support a standard that facilitates interoperability between applications, regardless of the platform on which they are run" (Short, 2002, p. 10).

A major contributing factor to the success of Web Services is that Web Services are built on existing, proven Internet standards such as XML and HTTP. This allows companies to leverage their existing infrastructure and investments they have already made in these technologies.

The following chapters will examine how Web Services can be leveraged for personal privacy protection.

Part II

WSP^3: The Model

Chapter 5

WSP^3: The Conceptual Model

In the first part of this study the issue of privacy and the implications of effective privacy support in service-based environments was examined. Chapter 2 provided an introduction to privacy as well as the most important issues and concepts surrounding it. An overview of current privacy legislation was also given. It was argued that an individual has a right to privacy and that it should be protected, especially on the Internet. Next, Chapter 3 looked at the various ways in which the computer industry has dealt with privacy issues. The mechanisms for increasing the level of individual privacy protection were also examined. Finally, Chapter 4 discussed service-based environments, focussing on Web Services and its supporting technologies.

The second part of this thesis sets out to develop a Web Service Model for Personal Privacy Protection, henceforth called the WSP^3 model or the model. A conceptual overview of this model will be given in the current chapter. Thereafter, the model will be elaborated in Chapters 6 and 7. Finally, Chapter 8 will present a prototype based on the model.

5.1 Scope Of The Model

From the previous chapters it seemed that privacy is a major concern, not only for individuals, but for businesses too.

Ideally, individuals would like to stay anonymous and not be identified. When providing identifying information users want to know that their information is protected, with some kinds of information being more sensitive than others (Cranor et al., 1999). To them, legislation and privacy protection

mechanisms are of prime importance.

Businesses are also facing new problems to cope with consumer privacy demands. They need to monitor existing legislation continuously and make sure a privacy policy is followed. In this regard the use of new technologies and security mechanisms are essential (Madsen & Adams, 2002).

The role of security is critical to a successful company, encompassing issues such as authentication, access control and cryptography to name but a few. Each of these concepts is relevant to privacy, such as providing confidentiality of information.

The role of information technologies has also increased dramatically. Technologies are the backbone of businesses, providing critical business functions and helping to ensure compliance with new legislation. In addition, novel technologies are continuously opening new business opportunities. Web Services is one such technology that promises much to businesses and individuals alike. Organizations are increasingly incorporating Web Services into their systems, promising easier business interactions and new customers experiences (Hallett, 2003).

Following from the discussion above, the scope of the model is set out in Figure 5.1 below.

The focus of the model falls in three areas, namely information technologies with the focus on Web Services, privacy and security. The implications of Web Services on the issue of privacy, and the security concepts involved are of primary concern. Thus, the primary focus of the model rests on the intersection of these three areas.

5.2 Conceptual Foundation

At a conceptual level the model is concerned with individuals, referred to as users, their private information and the Web sites they interact with.

The fundamental goal of the model is to provide users with a greater degree of control over their private information by defining a mechanism in which they are able to express the desired level of privacy for their information. In this regard no one should be able to obtain any information about a user without their explicit consent.

To support this aim users must be capable to assign a level of privacy to

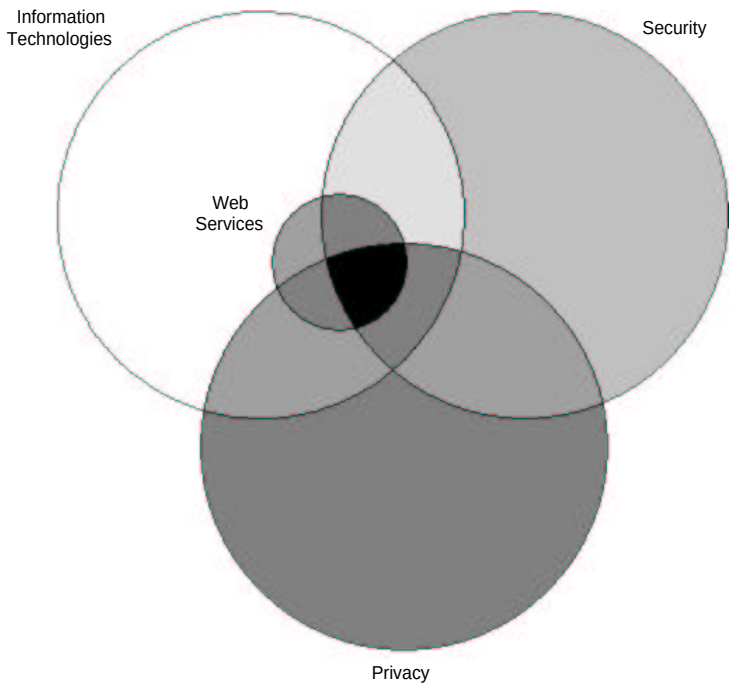

Figure 5.1: Model Scope

their private information. In addition, they must be able to control access to this information by assigning a level of trust to Web sites they interact with.

These concepts are based on the Hierarchical Privacy-Sensitive Filtering Model by Oberholzer (2001, p. 76–89). In this model a privacy sensitivity level is assigned to every data item. A user privacy-sensitive level is assigned to users who view the data. The principle of the model is that only content with a privacy level less than or equal to the viewer's level will be available to that viewer.

The enforcement of these privacy preferences will be implemented through Web Services. Using such a service-based model provides several supporting benefits. Compatibility between various technologies is promoted due to the platform-independent and non-proprietary nature of Web Services. Users lacking processing power benefit from the distributed computing environment.

5.3 Overview Of The Model

The WSP^3 model builds on these principles of privacy and trust levels, focussing on a Web Services environment. Figure 5.2, depicts the primary entities in the model and the interaction between them.

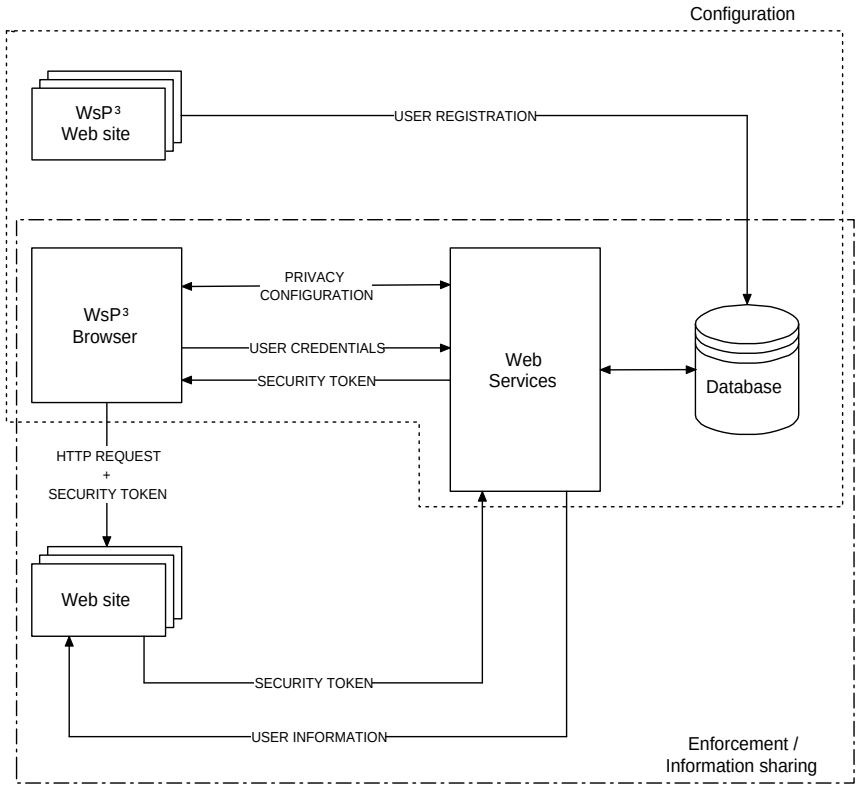

Figure 5.2: WSP^3 Model

As indicated, the model consists of two aspects: a configuration aspect and an enforcement/information sharing aspect.

The configuration facet defines the users and their privacy preferences. Thereafter the enforcement/information sharing side defines how private information is shared in accordance with these user preferences.

In the figure, the registration of users is performed through a Web site, storing the credentials directly in a central data repository. Users then configure their privacy preferences using a "WSP^3 sensitive" browser and a Web Service interface. The Web Service interacts directly with the data store and provides the interface to user information. When users wish to enforce their privacy preferences they authenticate themselves and receive a security token

in return. This security token accompanies requests to Web sites and is used by a Web site to request information about the user. The Web Service then returns information to a Web site based on the privacy preferences of the specific user.

5.4 Conclusion

The present chapter provided a conceptual overview of the WSP^3 model. In the next three chapters a detailed specification of the model will be given.

In order to allow private information to be shared in a secure manner, users need to specify their individual privacy requirements. Chapter 6 discusses this initial configuration process.

Thereafter, the sharing of private information according to the user's privacy preferences is presented in Chapter 7.

In Chapter 8 the theoretical aspects of the model are practically demonstrated through the development of a prototype based on the WSP^3 model.

Chapter 6

WSP^3: Setting The Scene

Chapter 5 provided a high level definition of the various components which comprise the WSP^3 model. This chapter will discuss in more detail the configuration aspect of the model.

For the the model to be effective some initial configuration and information capturing are required. This will allow each user to customize the environment to his or her own privacy requirements.

This chapter will examine the registration and configuration steps required in the initial setup process. These steps will allow future service authentication and information exchange with a Web site respectively.

6.1 Registration

An essential initial requirement of the model is user registration. Once completed, this will allow users to authenticate themselves to the service. Only authenticated users will be granted access to service resources, such as databases. Users will access the Web Service for configuration purposes, whereas Web sites will perform the interaction the rest of the time.

The registration process allows users to choose their service-login credentials, consisting of a unique username and a password. As usernames will be used to identify users, no duplicates are allowed. The interface for this registration process is provided through a Web site.

The user credentials are highly sensitive pieces of information which need to be kept private. For this purpose an encrypted communications channel is established between the client and server. This secures the clear text

credentials submitted via the registration process and assures users of the privacy and confidentiality of their information.

The Web site stores the user credentials in a database. Future authentication requests will validate user credentials against the database store.

A danger exists, when storing authentication information, that if a hacker manages to break into the database he or she can then use the usernames and passwords freely, possibly at other Web sites if users established the same credentials elsewhere. To prevent this, hashes of passwords are stored instead of clear text. The hash value is also combined with a salt value to mitigate the threat associated with dictionary attacks. The salt value is simply a random string of fixed size which is appended to the password. This prevents anyone from viewing the passwords, should they fall into the wrong hands.

Figure 6.1 provides a high level overview of the registration process and the security steps involved.

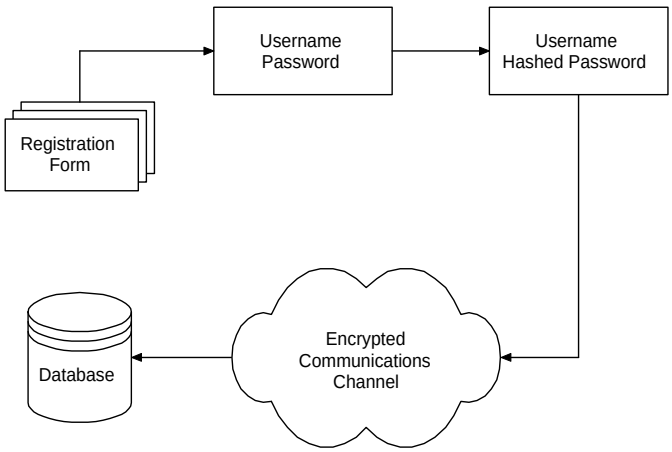

Figure 6.1: Registration Security

Once users have successfully registered they need to configure their privacy preferences using a browser. This browser will allow users to browse the Internet while keeping their personal information private. The following sections will discuss the configuration of an information profile and privacy preferences.

6.2 Information Profile

Users will configure their own profiles containing their personal information as well as their privacy preferences. A user is defined as follows:

Definition 6.1 (User). The set $USER = \{u_1, \ldots, u_n\}$ represents the collection of users in which each user represents an individual user. □

Thus, two separate users Bob and Alice will both set up their own personal profile and preferences.

Every user has the ability to exercise control over the storage of his or her personal information, or attributes. This allows users the convenience of a central repository of information which, when updated or changed, will automatically reflect changes to other parties as specified by the user. These attributes present information commonly requested by Web sites, such as name, address, contact information, credit card information, etc. Attributes are represented in the following way:

Definition 6.2 (Attribute). A set of attributes is stored, represented by $ATTR = \{a_1, \ldots, a_m\}$. □

Values for attributes come from a general domain of values, here represented as follows:

Definition 6.3 (Value). The value domain is the set $VAL = \{val_\varnothing\} \cup VAL_{all}$ where

- $val_\varnothing$ represents a null (or empty) value, and
- VAL_{all} represents all possible not null values. □

Although attributes apply to all users, a relationship exists between an individual user and an attribute. Thus Bob will give his phone-number attribute a specific value, while Alice will use another value. This relationship between users and their attributes is defined as follows:

Definition 6.4 (User Attributes). Each user has a set of attributes expressed as $USERATTR \subseteq USER \times ATTR$. □

Each attribute will contain a specific value, set according to the user's information. Considering the example from the previous paragraph, Bob's

phone-number might have the value 999 123 4567, while Alice's is 111 987 6543. This relationship between the user attribute and its value is defined as follows:

Definition 6.5 (User Information). The values assigned to user attributes are reflected in the $USERINFO$ relation, which is defined as

$$USERINFO \subseteq USERATTR \times VAL.$$

□

Once a profile has been established users can define the level of privacy they require for their attributes. This privacy configuration is discussed next.

6.3 Privacy Configuration

When configuring privacy the primary entities involved are the individual users, their personal information and Web sites. Additionally an associated value in the Personal Privacy Level (PPL) relation exists for each item of personal information, as well as a value in the Web site Trust Level (WTL) relation for each Web site.

At the heart of the model is the Personal Privacy Level. A user associates a value in the PPL relation with each of his or her attributes. This represents the user's level of sensitivity about disclosing that attribute. Thus a low value in the PPL relation would indicate that the information is not sensitive and may be communicated freely, while a high value in the PPL relation would indicate extremely sensitive data only available to selected parties. This relationship between information sensitivity and PPLs is illustrated in Figure 6.2.

This relationship between a PPL and an attribute is formalized as follows:

Definition 6.6 (Personal Privacy Level). A user assigns a value in the PPL relation to each attribute, expressed as

$$PPL \subseteq USERATTR \times \mathbb{N}$$

where $i \in \mathbb{N}$ represents the i'th level of sensitivity similar to the "privacy sensitivity levels" defined by Oberholzer (2001, p. 77). □

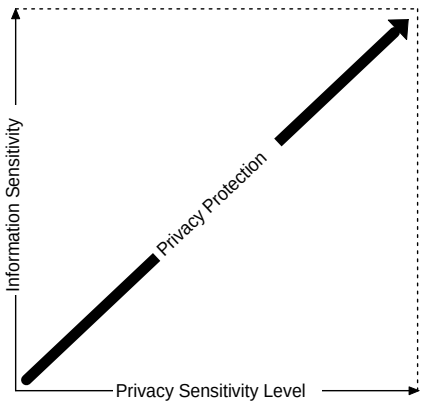

Figure 6.2: Privacy Protection

Thus Bob will assign a specific value to his phone-number attribute and decide to give it a 1 in the PPL relation. The resulting tuple, ((Bob, Phone-number), 1), indicates that he does not think the information to be very sensitive. Alice, however might assign a 3 in the PPL relation to her phone-number attribute. The tuple, ((Alice, Phone-number), 3), thus indicates a higher level of sensitivity and requires increased privacy protection.

To provide privacy protection while using the Internet, information regarding the Web sites that a user interacts with must also be stored. The Web Service must be aware of all Web sites subscribing to the service. These are represented as follows:

Definition 6.7 (Web site). The set $WEB = \{w_1, \ldots, w_x\}$ represents the collection of Web sites in which each individual Web site is unique. □

For each Web site a Web site Trust Level (WTL) is defined. This WTL is another core element of the model and is linked to the concept of a PPL.

Definition 6.8 (Web site Trust Level). A user assigns a value in the WTL relation to each Web site, expressed as

$$WTL \subseteq USER \times WEB \times \mathbb{N}$$

where $i \in \mathbb{N}$ represents the i'th level of sensitivity similar to the "user privacy-sensitivity levels" defined by Oberholzer (2001, p. 77). □

This value in the WTL relation represents the level of trust that exists between the user and a specific Web site. A Web site which the user knows

well and trusts may be given a high value in the WTL relation, while unknown sites will be given a low value in the WTL relation. For example, Bob might trust Microsoft completely and assign their Web site a 3 in the WTL relation, forming the resulting triple (Bob, Microsoft, 3). Alice might only feel a moderate amount of trust for Microsoft and assign them a 2 in the WTL relation, forming the triple (Alice, Microsoft, 2).

The key principle of the model is that a Web site may only view attributes about a user where the attribute's value in the PPL relation is *less than or equal to* the Web site's value in the WTL relation. This relationship is illustrated in Figure 6.3.

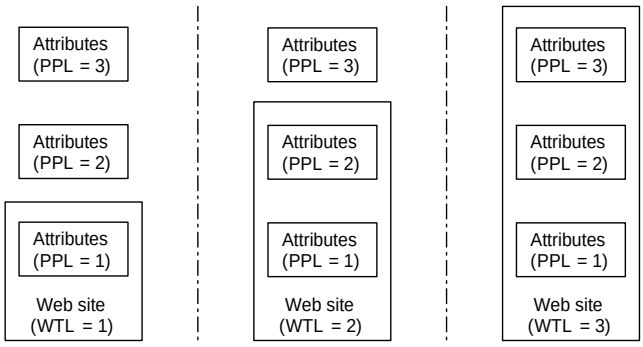

Figure 6.3: Viewable Attributes

When considering the examples given above, Bob's phone-number attribute, assigned a 1 in the PPL relation by himself, will be available to Microsoft, assigned a 3 in the WTL relation, to view. However, Alice's phone-number, assigned a 3 in the PPL relation, will not be available to Microsoft, as she only assigned a 2 in the WTL relation to the Web site. How this is resolved is formalized in Chapter 7.

The privacy configuration aspect of the model, described in this section, is provided through a browser. This provides the interface through which users can access and modify their personal information and configure their PPL's and WTL's. The information is stored in a database via a Web Service, allowing users to access their information from any location. Information about the Web sites that users wish to interact with are also stored. This information includes the full URL and is used when exchanging information.

The next section will discuss how the privacy of a user's information is to be ensured during the information profile and privacy configuration steps.

6.4 Securing Communications

As all information about the user is of a highly confidential nature, a secure channel of communication is needed between the browser and the Web Service. This will prevent any intercepted data from being viewed. Communications between the browser and Web Service are secured through encryption. For this purpose both symmetric and asymmetric encryption are used.

A session is started every time a user performs a service-login using the browser. Along with the user credentials a random session key is generated which will be used to encrypt further communications between the browser and Web Service. The session key is defined as follows:

Definition 6.9 (Session Key). A new session is started every time the user performs a service-login and lasts until the user logs out. For the duration of this session a randomly generated symmetric key ($K_{session}$) will be used for encrypting communications between the browser and Web Service. □

As the Web Service does not yet have the session key it is communicated to the service securely by encrypting it with the public key of the service ($P_{service}$). The service can then decrypt the message using the corresponding private key ($P'_{service}$). This message is illustrated below:

Message 6.1 (Sharing the Session Key). The expression

$$Service\text{-}login = (Username, Password, K_{session})P_{service}$$

represents a service-login, where

- $Username$ and $Password$ are the credentials of the user to be authenticated against the database store,
- $K_{session}$ is the randomly generated session key, and
- $P_{service}$ is the public key of the Web Service. □

If a user is successfully authenticated the Web Service stores the session key along with a unique session-id for future use. This session-id is communicated to the browser for future messages in the following way:

Message 6.2 (Sharing the Session-id). The session-id is encrypted using the session key and sent to the browser which is the only party with a corresponding session key to decrypt the message.

$$(Session\text{-}id)K_{session}$$

□

As many users may be using the service at the same time it is necessary that encrypted messages from the browser also contain the correct session-id so that each message can be successfully matched to the correct user and decrypted. This is illustrated as follows:

Message 6.3 (Secure Communications). All communications between the browser and service are encrypted using the session key. The session-id accompanies each message to allow successful decryption.

$$Secure\ Message = ((Information)K_{session}, Session\text{-}id)$$

□

The messages between the user and the Web Service to establish a secure communications channel are illustrated in Figure 6.4.

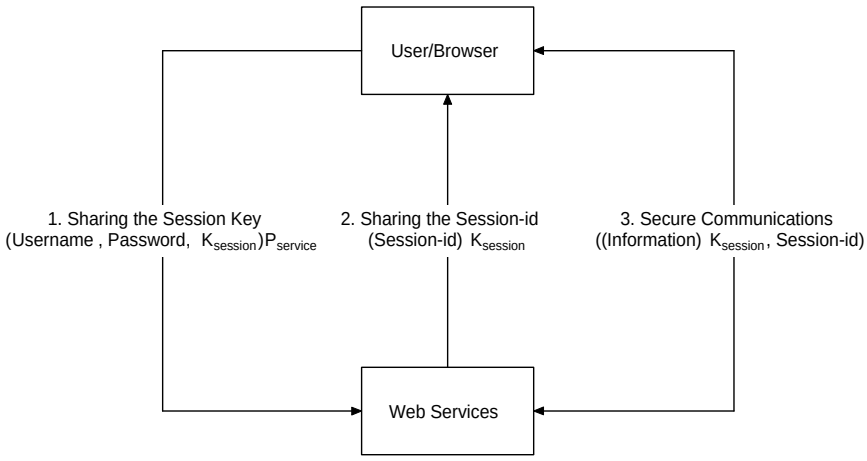

Figure 6.4: Securing Communications

When the session is ended by the user the session key is destroyed by both the browser and Web Service and cannot be used again. This mechanism ensures that user information stays private while being exchanged with the services.

6.5 Conclusion

This chapter has defined the initial steps required by the model as well as the mechanism for securing communications between the browser and Web Service. Users need to register their credentials for future authentication to the services. A personal information profile is to be completed as well as the assignment of PPLs to attributes and WTLs to Web sites.

The browser allows users to perform these steps while maintaining information privacy and security. Communications between the browser and services are secured through encryption.

The next chapter will examine the model in the context of secure information exchange with Web sites using the Web Service.

Chapter 7

WSP^3: Sharing Private Information

Chapter 6 discussed the configuration steps needed to ensure that the model functions correctly. Of primary importance was the configuration of privacy settings for user information as well as Web sites that a user will visit.

This chapter defines the model in the context of exchanging user information with Web sites. This must be done according to the user's privacy preferences and at all times be communicated securely to ensure continued information privacy.

Following, the mechanism for identifying users to Web sites are defined. The interaction between a user, a Web site and the Web Service, as well as the sharing of user information with a Web site is formalized.

7.1 State Maintenance

Because users are unlikely to browse the Internet in one continuous session, a mechanism is needed to keep track of users through each online session. This will identify users to the Web Service and allow Web site requests for visitor information to be matched to the correct user.

A security token, generated by the Web Service, identifies each user during a session and is defined as follows:

Definition 7.1 (Security Token). A unique security token is generated each time a user is successfully authenticated for a new session. This token

identifies the user and contains a time-stamp to keep track of the current session length:

$$\textit{Security-token} = (Username, \textit{time-stamp})P_{service}$$

The token contents is secured by encrypting it with the public key ($P_{service}$) of the Web Service. □

Only the Web Service must be able to link this token to a user. Because the token is encrypted using the public key ($P_{service}$) of the Web Service, it will be the only party with a matching private key ($P'_{service}$) to decrypt the token. Additionally the security token must also keep track of the length of the current session. If the session length exceeds a predetermined length of time the token will become invalid and another one must be requested from the Web Service. These criteria will ensure that:

- no other entities, such as Web sites, will be able to identify a user based on his/her token, and
- interception of a token and subsequent masquerading will be kept to a minimum as tokens will have a limited lifetime.

Once a security token has been generated by the Web Service and received by the user, it accompanies every request to a Web site. If the token should expire before the session has ended, a new token is generated and used in the same way.

7.2 Web site Interaction

Because a security token identifies each user, it must accompany each request to a Web site. The Web site is then responsible for examining the page requests and extracting the security token from the request.

To query the Web Service for the attributes of the user associated with the token, the Web site must send a request to the Web Service which includes the token. Additionally the Web site must also include some identification information so that the message can be authenticated by the Web Service. Thus the request will look as follows:

Message 7.1 (Information Request). The expression

$$Information\ Request = (Security\text{-}token, Site\text{-}identification)$$

represents the information request from a Web site, where

- *Security-token* is the token identifying the user, and
- *Site-identification* represents some authentication data which will uniquely identify the Web site to the Web Service. □

This will ensure that only consented information is transmitted to parties/Web sites. The resulting interaction between the Web Service, user and Web site regarding the security token is shown in Figure 7.1.

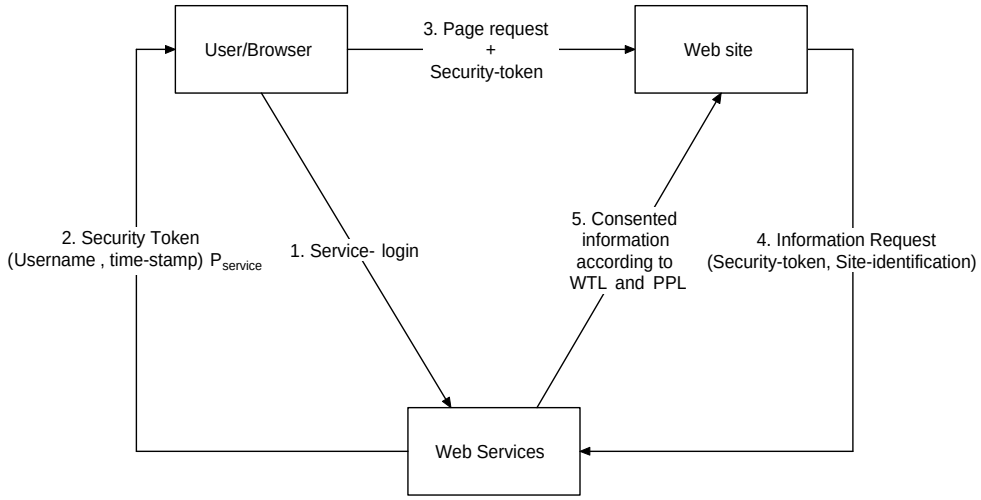

Figure 7.1: Security Token Interaction

Once the Web Service is satisfied that the message is authentic and that the token has not expired, the request can be processed.

7.3 Information Sharing

Once a Web site has received a security token and has requested the user's attributes, the Web Service needs to determine the information available to the Web site based on the privacy settings of the relevant user.

This is done by comparing the value in the Web site Trust Level (WTL) relation of the Web site making the request to the value in the Personal Privacy Level (PPL) relation of each user attribute. Determining the value

of the attribute to be released to the Web site is done according to the following definition:

Definition 7.2 (Get Attribute). The function *GetAttribute* is defined as

$$GetAttribute: USER \times WEB \times ATTR \rightarrow VAL$$

where $GetAttribute(u, w, a) =$

- v if $\exists\, ppl \leq wtl$ such that
  $((u, a), v) \in USERINFO \wedge$
  $((u, a), ppl) \in PPL \wedge$
  $(u, w, wtl) \in WTL$
- $val_{\varnothing}$ otherwise. □

According to this definition an empty value is returned to the Web site if its value in the WTL relation does not match or does not exceed the attribute's value in the PPL relation, otherwise the attribute is returned to the Web site.

To ensure that the user attributes remain confidential, all communications back to a Web site need to be secured. This is done by encrypting the messages with the public key of the Web site (P_{site}). Only the Web site will then be able to decrypt the message using the corresponding private key (P'_{site}). This message is illustrated below:

Message 7.2 (Securing Web site Communications). The return message is encrypted using the public key (P_{site}) of the Web site. It is transmitted to the Web site which is the only party with a corresponding private key (P'_{site}) to decrypt the message.

$$(Information)P_{site}$$

□

This will ensure that only the intended recipient (the requesting Web site) is able to view the information.

7.4 Conclusion

This chapter has defined the mechanism for sharing private information. Central to this mechanism is the security token identifying each user and the comparison of privacy levels, used to determine which information can be shared.

In combination with the previous chapter the complete model has been defined and the mechanisms underlying it explained.

In the next chapter a prototype based on the model will be examined. This prototype will implement the configuration steps, security features and information exchange mechanisms essential to the model.

Chapter 8

WSP^3: Prototype Implementation

This chapter concludes the second part of this dissertation by presenting a prototype implementation of the WSP^3 model. This serves to provide a clearer understanding of the different aspects of the model as well as a proof-of-concept.

The prototype consists of various components, each implementing a specific part of the model. These components can be roughly divided into the following:

- The primary part of the prototype consists of a set of Web Services which provides the interface for client interaction as well as the enforcement of the model's principles.
- A Web site allows users to register their credentials prior to using the service.
- Thereafter a browser allows users to configure their personal information and privacy preferences as well as interact with Web sites.
- Lastly a sample Web site illustrates the implementation of the user's privacy preferences.

This chapter will examine each of these components and discuss the operation of the prototype. Before this, however, an overview of the development tools is given.

8.1 Choice Of Tools

The prototype was developed using the Microsoft .NET Framework version 1.1. Apart from providing the ability to create Web Services, Web applications and desktop applications, the framework also provides an extensive library of supporting classes. These classes were used extensively throughout the prototype and will be discussed where applicable.

All components of the prototype were programmed using Visual C# .NET and the Microsoft Visual Studio .NET 2003 integrated development environment (IDE). For data storage Microsoft SQL Server 2000 was used.

The following sections will discuss the various components of the prototype developed using the above tools, starting with the Web Services.

8.2 The Web Services

At the heart of the model a family of Web Services provides an interface through which all information is managed. These services allow users to access and maintain their information and privacy settings. In addition, the services manage requests for private information from Web sites, implement privacy preferences and return the relevant information.

To provide the aforementioned functionality the Web Services rely on various classes and "helper" functions. Amongst other things, these classes and functions encapsulate various messages travelling between the Web Services and clients. Additionally, they provide database access and encryption routines to secure communications.

The cryptographic functions are particularly important for ensuring the privacy of messages between the user and Web Services. The *Crypto* class contains cryptographic functions for generating a symmetric key, encryption and decryption. The function which encrypts sensitive information is shown below:

```
1 public byte[] Encrypt( byte[] plainText, byte[] key )
2 {
3     SymmetricAlgorithm alg = SymmetricAlgorithm.Create( "Rijndael" );
4     alg.Mode = CipherMode.ECB;
5     alg.KeySize = 128;
6     alg.Key = key;
7
```

```
8       MemoryStream ms = new MemoryStream();
9
10      CryptoStream cs = new CryptoStream(
11          ms, alg.CreateEncryptor(), CryptoStreamMode.Write );
12
13      cs.Write( plainText, 0, plainText.Length );
14      cs.Close();
15
16      byte[] encryptedData = ms.ToArray();
17
18      return encryptedData;
19  }
```

Function *Encrypt* receives a byte array of plaintext to encrypt and the key to use. Firstly, a new instance of the "Rijndael" symmetric algorithm is created (line 3). Next, lines 4–6 configure various aspects of the algorithm and sets the key used by the algorithm. On line 13 the actual encryption is performed by writing the plaintext input to the *CryptoStream* declared on line 10. Finally the encrypted data is converted to a byte array and returned (lines 16 and 18).

The decryption functions which reverses the above process is illustrated below:

```
1   public byte[] Decrypt( byte[] cipherText, byte[] key )
2   {
3       SymmetricAlgorithm alg = SymmetricAlgorithm.Create( "Rijndael" );
4       alg.Mode = CipherMode.ECB;
5       alg.KeySize = 128;
6       alg.Key = key;
7
8       MemoryStream ms = new MemoryStream();
9
10      CryptoStream cs = new CryptoStream(
11          ms, alg.CreateDecryptor(), CryptoStreamMode.Write );
12
13      cs.Write( cipherText, 0, cipherText.Length );
14      cs.Close();
15
16      byte[] decryptedData = ms.ToArray();
17
18      return decryptedData;
19  }
```

Function *Decrypt* is almost identical to the *Encrypt* function above. It takes as parameters a byte array of ciphertext and the key to use for decryption. The only difference is on lines 10 and 11 where *alg.CreateDecryptor* is

used instead of the corresponding encrypter. Finally, lines 16 and 18 return the original byte array of data.

The Web Services also define an asymmetric key pair for the initial key exchange (message 6.1). For the prototype the key pair is stored inside the class definition as an Extensible Markup Language (XML) string. In a production system this would rather be implemented using an X.509 digital certificate, thus hiding the private key and providing more security.

As the various Web Services are all called by the browser application and Web sites, they will be discussed in the appropriate sections where relevant. The next section will discuss the registration Web site and the registration process.

8.3 The Registration Web site

The registration Web site implements the required functionality as set out in section 6.1. As all users of the service need to have Internet access a Web site is a very convenient implementation choice. Figure 8.1 illustrates the Web page on which registration takes place.

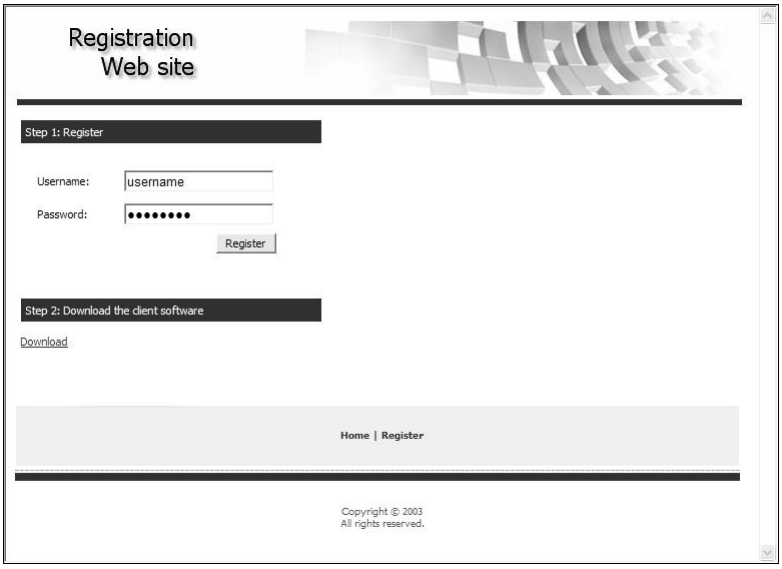

Figure 8.1: Registration Web page

As can be seen in the figure, registration is a two step process.

Firstly, users need to choose a username and password for future authentication to the service. As stated in section 6.1, usernames will be used to

identify users and must therefore be unique. This uniqueness is enforced by a stored procedure, illustrated below, before a username is committed to the database:

```
1  IF (SELECT count(*) FROM [User] WHERE userName = @userName) > 0
2    BEGIN
3      -- User already exists
4      RETURN (1)
5    END
6  ELSE
7    BEGIN
8      INSERT INTO [User] (userName, passwordHash, salt)
9      VALUES(@userName, @passwordHash, @salt)
10     RETURN (0)
11 END
```

The stored procedure checks whether a user has already been registered with the same name and if so returns 1 (line 4), indicating an error. Otherwise the user credentials are stored in the database (lines 8–10).

As stated in section 6.1, passwords are highly sensitive pieces of information and should not be stored in clear text. Instead a hash of a password combined with a salt value should be stored. The .NET Framework provides the classes necessary to accomplish these requirements as is illustrated below:

```
1 private string CreateSalt( int size )
2 {
3     RNGCryptoServiceProvider rng = new RNGCryptoServiceProvider();
4     byte[] buff = new byte[size];
5
6     rng.GetBytes( buff );
7
8     return Convert.ToBase64String( buff );
9 }
```

The *CreateSalt* function receives a number representing the size of the hash as parameter and returns a string representation of the salt value. On line 3, an object of the *RNGCryptoServiceProvider* class is declared which implements a cryptographic Random Number Generator (RNG). An array of bytes is declared on line 4 to hold a salt value of the given size. Method *GetBytes* on line 6 then fills the array of bytes with a cryptographically strong random sequence of values. This sequence is then converted to a string and returned (line 8).

After a salt value has been calculated the password hashing is performed. A hash function is a cryptographic algorithm which produced a different

output for each unique element of data. Thus two identical passwords have the same hash but in this case, because a random salt value is added each time, each hash is unique. The code below shows this process:

```
1  private string CreatePasswordHash( string pwd, string salt )
2  {
3      string saltAndPwd = String.Concat( pwd, salt );
4
5      string hashedPwd =
6          FormsAuthentication.HashPasswordForStoringInConfigFile(
7          saltAndPwd, "SHA1" );
8
9      return hashedPwd;
10 }
```

Function *CreatePasswordHash* accepts the password and salt value as parameters and return the hashed string. On line 3, the password and salt value are first concatenated before the hashing is performed. The *HashPasswordForStoringInConfigFile* method of the *FormsAuthentication* class is then used to generate the hash value using the given algorithm, in this case "SHA1". Finally line 9 returns the hashed password which is then stored in the database along with the username and salt value. The salt value needs to be stored as well to allow passwords to be compared with hashed values later.

One point to note using such an implementation is that if a user forgets his or her password there is no way of getting it back. To alleviate such occasions the use of a password reminder is recommended. If this is also forgotten the user will need to register again. These factors could result in the size of the database being inflated by "unused" users.

A secure communications channel between a client and the Web site is established using Secure Sockets Layer (SSL). This requires that an X.509 digital certificate is installed on the server hosting the Web site. This certificate should be signed by a Certificate Authority (CA) to be considered trustworthy. After installation the client can ascertain whether the connection is secure by referring to the URL which should start with *https://*.

The second part of the registration process requires users to download and install a custom browser which will allow a profile and privacy preferences to be established. This is discussed in the next section.

8.4 The Browser

The browser is a basic application for viewing Web sites, but also allows users to maintain an information profile and configure their privacy preferences. Figure 8.2 shows the main screen of the browser.

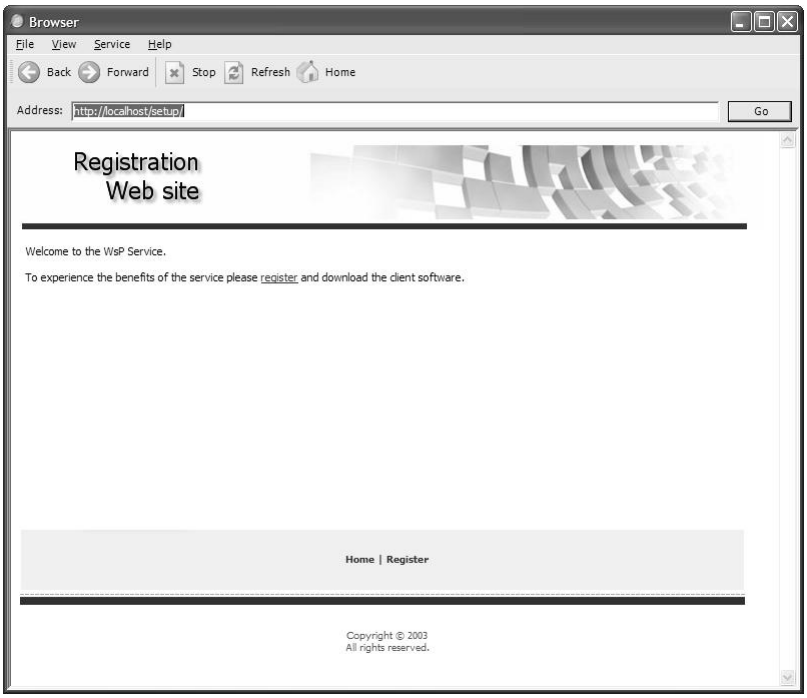

Figure 8.2: Browser

When using the browser the user needs to authenticate him or herself to the Web Services by supplying the credentials chosen during registration. This is done by choosing *Login* from the *Service* menu. Figure 8.3 illustrates the login window.

Figure 8.3: Browser Login

As stated in definition 6.9, the browser generates a unique session key when a user logs in. This is done using the *Crypto* class referred to ear-

lier. The prototype implements the sharing of the session key (message 6.1) slightly different as defined. Rather than encrypting the username, password and session key using the public key of the service, it encrypts and sends only the session key. This is necessitated by the asymmetric encryption classes in the .NET Framework, which are designed to encrypt only a small amount of data such as cryptographic keys.

Also, instead of using a session-id to identify users (message 6.2), the prototype uses an *HTTPSessionState* object to keep track of each user's session. This eases session tracking by eliminating the need for a session-id, but requires that cookies be enabled on the client machine.

The session key is used to encrypt all further communications between the browser and Web Services. Once the session key has been exchanged, the browser encrypts the user credentials and sends them to a Web Service for authentication. If a user is valid, a security token is returned. This token (definition 7.1) consists of an encrypted *SecurityToken* object, encapsulating the username and a time-stamp. Because of the asymmetric encryption limitation mentioned above, the security token is encrypted using a secret symmetric key known only to the Web Services, rather than the service public key. The security token is stored by the browser for future Web site interaction as discussed in section 7.2.

After the authentication process the user should configure an information profile and privacy preferences as described in sections 6.2 and 6.3. This is done by the choosing *Options* from the *Service* menu, shown in Figure 8.4.

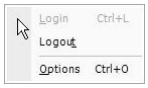

Figure 8.4: Service Menu

Here the user can capture values (definition 6.3) for the defined attributes (definition 6.2) which are then stored in a database via a Web Service. For the purposes of the prototype personal information is grouped into various categories, roughly based on the Windows Address Book. Although this is different from the model, it merely serves to ease the administration of the Personal Privacy Level (PPL) relation and does not detract from the generality of the model. Users must also assign a value in the PPL relation,

as described in definition 6.6, to each attribute. For manageability attributes have been grouped together and each attribute of the same category receives the same value in the PPL relation. Figure 8.5 illustrates the user attributes stored in the name category and their PPL.

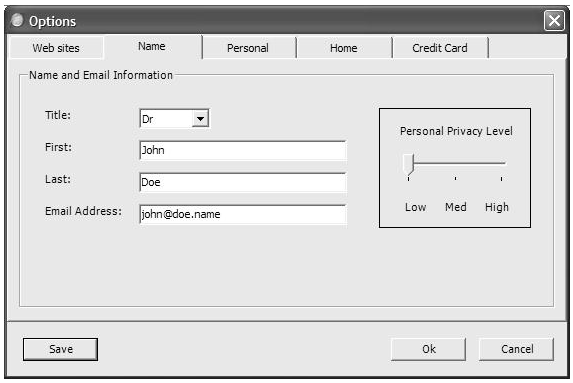

Figure 8.5: User Information

Information about the Web sites a user wishes to interact with are also stored using the options screen. Figure 8.6 shows the Web sites tab.

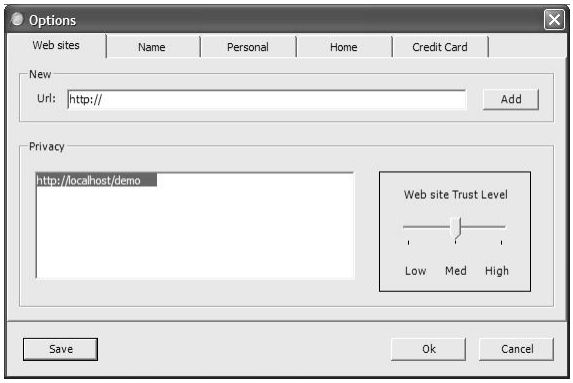

Figure 8.6: Web site Information

As can be seen in the figure, users can add Web sites and assign a value in the Web site Trust Level (WTL) relation, as described in definition 6.8, to each site.

Personal information is stored in a database through a Web Service interface. When the options form is opened the Web Service returns an encrypted *User* object which encapsulates all user information. Any changes to the information is updated by another Web Service using the same encrypted object. The same concept applies to Web site information which is

encapsulated in a *WebSite* object. In this manner the privacy of personal information is ensured even if the communications between the browser and Web Services is intercepted.

After the configuration process is complete the user can use the browser to navigate participating Web sites, as defined on the Web sites tab, privately. As stated in section 7.2 the security token accompanies each request to such a site. The browser sends the token in the HTTP header collection with each request. The code to accomplish this is shown below:

```
1  private void Browse( Object oUrl )
2  {
3      try
4      {
5          Object oEmpty = "";
6          Object oHeader = "";
7
8          if ( securityToken != "" )
9              oHeader = "WsP: " + securityToken + "\n" + "\r";
10
11         axWebBrowser.Navigate2(
12             ref oUrl, ref oEmpty, ref oEmpty,
13             ref oEmpty, ref oHeader );
14     }
15     ...
16 }
```

Method *Browse* receives an object representing the Uniform Resource Locator (URL) to navigate. Line 8 checks for the existence of a security token. If it exists it is added to the header collection along with the required string formatting. On lines 11–13 the actual navigation is performed. Figure 8.7 illustrates the request resulting from the above code when a security token exists.

```
GET /demo HTTP/1.1
Accept: */*
Accept-Language: en-gb,af;q=0.5
WsP:
KOVAbUlBZVNzPSGrKU5EgoWVFq3NNAqEk3JVRabN4rLHQ+Je5yLA1KG1JP3AyOb8Kvm
P+6OmMeCH8zLteHurC1ZPx/c84XJ/Xjt+2pfYDERD5Wfkmoxm6i8fpByozfepz16Znag7LQIEvs
M8nQDl4pkHLFlf1l1n3xBueMKdcuoZot8/JXDjdhvbTD7ozGnPaKy3Eaqlfg1NLvTVdF8rPx3Og1j
O5KuDa8SSx3DvPUQL4c4+S/HyJuyw7ZxU76Rw
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; .NET CLR 1.1.4322)
Host: localhost:8080
Connection: Keep-Alive
```

Figure 8.7: HTTP Request

Once a Web site receives such a request it is responsible for further pro-

cessing. When a user is finished with the current session he or she can log out of the service. This destroys the security token and thus ends the session. The next section examines the processing of a security token by a Web site.

8.5 The Demonstration Web site

The demonstration Web site illustrates the end result of the model: information being shared according to a user's privacy preferences. It serves to show what happens when no security token is received, when a token has expired and how access to information differs when users change their PPL and WTL preferences.

To distinguish between users of the service and non-service users the Web site continuously checks the HTTP headers for a security token. This is a simple process, illustrated below:

```
1 if (Request.Headers["WsP"] != null)
2 {
3     string securityToken = Request.Headers["WsP"].ToString();
4     ...
```

Line 1 above checks for the existence of a "WsP" HTTP header. If such a header exists the contents of the header is retrieved and stored (line 3).

If a token is available the user's information can be queried through a Web Service. As defined in message 7.1, such a request must contain the security token and some information identifying the Web site. For the prototype the site-identification is simply a string containing the site URL. The string must match exactly the one to which the user assigned a value in the WTL relation using the browser. Once again a more secure implementation would be the use of an X.509 digital certificate. This would be used to check the authenticity of the Web site, as well as encrypting the return message from the Web Service, using the Web site's public key.

The Web Service receiving the information request, checks the validity of the token, retrieves the relevant user information, and returns it to the Web site.

Firstly, the token needs to be decrypted and the time-stamp checked. This ensures that intercepted tokens cannot be used indefinitely for masquerading purposes. In the prototype all tokens expire after five minutes. In normal

circumstances the time before a token expires would be increased to a suitable level. If a token has expired an error message is returned to the Web site and a new token must be obtained by the user by logging in to the service again.

If a token is valid the Web Service builds an XML string containing the information available to the Web site. This is done as set out in section 7.3 and is partly illustrated below:

```
1  ...
2  sql += "SELECT ";
3
4  for (int i = 0; i < reader.FieldCount; i++)
5  {
6      if ( !reader.IsDBNull(i) )
7      {
8          if ( reader.GetInt32(i) <= wtl )
9          {
10             sql += reader.GetName(i) +
11                 " AS " +
12                 reader.GetName(i).ToUpper();
13         }
14         else
15         {
16             sql += "' ' AS " +
17                 reader.GetName(i).ToUpper();
18         }
19         columnAdded = true;
20     }
21
22     if ( columnAdded )
23     {
24         if ( i < reader.FieldCount - 1 )
25         {
26             sql += ", ";
27         }
28         columnAdded = false;
29     }
30 }
31
32 sql += " FROM [User] WHERE userName = '" +
33     userName +
34     "' FOR XML AUTO, ELEMENTS";
35 ...
```

The above snippet of code builds a Structured Query Language (SQL) string using the values in the PPL and WTL relations specified by the user. The variable *sql* holds the string, while the variable *wtl* (line 8) has been

assigned the WTL of the Web site making the request. In the prototype a WTL value of one is assumed if the site-identification cannot be matched exactly. Lines 4–30 loop through the user's PPL settings. Line 8 compares each PPL value to the WTL value of the site, retrieving the column value or an empty value as appropriate. Lines 22–29 make sure that the SQL is well-formed.

The query is then executed and the resulting data is encrypted before being sent to the Web site. Figure 8.8 shows a typical example of such an information request.

Figure 8.8: Information Received

Once the information has been received from the Web Service and decrypted, the Web site can format and utilize it according to the privacy policy of the Web site.

8.6 Observations

In the development of the prototype it was aimed to implement each of the features required by the model as they were defined. Some features were,

however, implemented differently due to shortcomings in the available classes or programming ease. The implications of these changes are examined in the following paragraphs.

Firstly the advantage of using Web Services as a base for the model allows a single sign-on environment to be created. Users are assured that all Web sites they visit will only be able to collect the information they have specified. Conversely, Web sites need to be configured to receive security tokens and process them in the appropriate manner.

Secondly, the use of cookies to keep track of session state greatly eases programming and reduces the amount of traffic flow between the browser and Web Services. This presents a definite gain in performance but has the drawback that users not willing to accept cookies cannot use the service.

Concerning the assignment of a value in the PPL relation to an attribute, the decision to group attributes and only use one PPL value for a group eases programming and creates a more manageable user interface. If the flexibility this approach lacks is a major concern, a PPL value for each individual attribute would have to be implemented.

Finally, the prototype does not make use of X.509 digital certificates. This is mainly because of the costs involved. To establish greater security it would be required that the Web Service *as well as* all participating Web sites have such a certificate. This would not only improve site authentication but may also be used to encrypt communications between the Web Service and Web sites. Web sites that do not already have such a certificate should carefully consider obtaining one in order to establish greater security.

8.7 Conclusion

This chapter demonstrated how the WSP^3 model could be implemented as a prototype. The prototype consisted of various components, each implementing a specific part of the model.

The registration Web site implemented the initial registration of users prior to using the service. Thereafter the custom browser allowed users to set up a profile and configure their privacy preferences. A demonstration Web site illustrated the retrieval of information according to the privacy settings of the user. At the heart of the prototype a set of Web Services provided an

interface for information management as well as enforcement of the model's principles.

The next chapter will conclude this dissertation by summarizing the contributions of the research.

Part III

Epilogue

Chapter 9

Conclusion

This dissertation presented the WSP^3 model for personal privacy protection in a Web Service environment. The model came about as the result of a thorough and systematic investigation into the subject matter as set out in section 1.4.

The set of realizations reported in Chapter 1 formed the motivation for this study. The problem area was then defined in section 1.2 in the form of a set of research questions. After a more detailed study into the research domain it was possible to formulate a model which supports personal privacy protection in a Web Service environment.

This chapter will discuss to what extent the WSP^3 model meets the objectives set in section 1.3. Issues in the implementation of the model will be highlighted and further research suggested.

First, however, a summary will be given of how the research questions, posed as part of the problem statement in section 1.2, were addressed.

9.1 Revisiting The Problem Statement

This dissertation addressed the issue of how Web Services could be leveraged to provide the individual with effective personal privacy protection.

To arrive at a conclusive answer to this problem a number of sub-problems were posed concerning the required level of privacy and the role of privacy in the Web Services architecture and its supporting technologies.

This dissertation attempted to answer these research questions in the following way.

9.1.1 Ascertaining The Individual's Required Privacy Level

It was established that every individual has his or her own perception of privacy and unique privacy preferences. A set of users might have very divergent perceptions about the importance of attributes common to them all.

It is thus the responsibility of the individual to ascertain the desired level of privacy for attributes belonging to him or her. One of the primary goals of the WSP^3 model is to allow users to select their privacy preferences in an intuitive, subjective, and consistent manner.

9.1.2 Privacy Using The Web Services Architecture

Web Services were thoroughly examined in Chapter 4. The various supporting technologies were discussed as well as the benefits to be gained.

Web Services is a relatively new technology, developing and improving at regular intervals. Ensuring the privacy of messages between clients and Web Services is of primary concern. Privacy is mainly achieved through security techniques, such as encryption, which secures the channel between the client and the Web Service.

The WSP^3 model makes extensive use of cryptography (for further information on cryptography see Appendix A) to provide privacy support in a Web Service environment.

9.1.3 Privacy In Web Service Supporting Technologies

The Extensible Markup Language (XML) was identified as the main supporting technology in the Web Services architecture. Almost all Web Service related technologies are XML-based.

By enabling smart data transfers of structured documents with logically identifiable content, XML contributes to the problem of privacy. However, XML-based efforts also offer potential applications to address various aspects of privacy. One such identified application (section 3.3) is the Platform for Privacy Preferences (P3P), the most visible industry supported privacy tool.

In the WSP^3 model the protection of information contained in XML com-

munications is achieved by employing encryption. This allowed the benefits of XML to be leveraged while adding a level of privacy support.

The next section will consider to which extent the proposed WSP^3 model has met the stated objectives in section 1.3.

9.2 Meeting The Desired Objectives?

Section 1.3 presented three desired features of a service-based privacy protection model. The following paragraphs will evaluate to what extent the WSP^3 model attained these requirements.

9.2.1 Specifying The Level Of Privacy

To allow users to specify their required level of privacy, section 6.3 defined a Personal Privacy Level (PPL) for each item of information (or attribute), and a Web site Trust Level (WTL) for each Web site visited by the user. The WSP^3 model is based on the key principle that a Web site may only view attributes about a user where the attribute's value in the PPL relation is *less than or equal to* the Web site's value in the WTL relation.

These definitions provide the means by which a user can select a protection level he or she is comfortable with for each attribute. In addition, these definitions established the mechanism by which the information will be shared later with Web sites, while maintaining the user's privacy preferences.

9.2.2 Implementing Privacy Using The Web Services Architecture

The WSP^3 model uses Web Services and its supporting technologies to establish an environment in which the privacy of the user is protected. The model employs encryption to secure private information travelling between the client and Web Service. The use of both asymmetric and symmetric encryption ensures that privacy is maintained throughout the model.

9.2.3 Integration With Current Standards

The WSP^3 model integrates well with current standards such as XML. Using these standards as a basis the model builds on them to provide privacy support. This ensures that the benefits of standard technologies are maintained while providing the intended privacy protection.

The implementation issues arising from these objectives are discussed in the following section.

9.3 Implementation Issues

The use of encryption worked well in the model because control over both the client as well as the server code was available. However, no standard way exists to encrypt data or to ensure message integrity. This implies serious concerns in a situation where any client should be able to use the service. Fortunately, current research should address this issue in the near future.

Although not employed by the prototype, XML Encryption (Eastlake, Reagle, Imamura, Dillaway, & Simon, 2002) could also be considered as an alternative encryption technique. The advantage of this approach would be the ability to encrypt only certain parts of a document, minimizing encryption processing and allowing non-sensitive information to be processed.

A philosophical issue arising from the model is whether an individual would be willing to entrust his or her personal information to a single entity (Web Service). In this matter , technology plays a lesser role to issues such as trust, privacy policies, and protective legislation.

9.4 Further Research

Although the WSP^3 model met the stated objectives, the following points need to be taken into consideration for the model to be a truly global solution:

- A problem arises if a user trusts a Web site but not all sub-domains of that site. In this case a system of partitioning Web sites needs to be implemented.

- By allowing users to assign different levels of privacy to their information for different Web sites, the model's flexibility and the user's

privacy is enhanced.

In addition to these considerations the work contained in this dissertation could be extended by future research initiatives. Some areas regarding such research are:

- The WSP^3 model did not address any issues arising from an information capturing perspective. Automating the process of assigning values in the PPL and WTL relations could possibly initiate another research project.
- Incorporating a Web site privacy policy check, similar to P3P (section 3.3), could be incorporated. Extending this mechanism to check for changes in privacy profiles since the last value in the WTL relation assignment may be investigated.
- Formulating a business model with regards to a payment scheme for use of the service may yield further research opportunities.
- The legal implications of implementing the model have not been examined here and is a possible topic for future investigation.

9.5 Epilogue

The author trusts that this work has raised the awareness of the issues in the domain of discourse. It is also hoped that the possibilities presented by new technologies to innovate as well as improve existing ways have been shown. The author would like to conclude this research by expressing the hope that this work will stimulate further work in the subject area.

Part IV

Appendices

Appendix A

Cryptography

Although cryptography does not fall directly under the scope of this study it features prominently as a security mechanism throughout.

Cryptography is a broad subject with a long and fascinating history. The details and subtleties inherent to the subject make a complete examination here impossible. Rather, this appendix provides a brief review of cryptography, focusing on symmetric and asymmetric encryption and the cryptographic support in the .NET Framework.

A.1 What Is Cryptography?

Cryptography is the science of encrypting and decrypting data using mathematics. It enables you to store or transmit sensitive information so that it cannot be read by anyone but the intended recipient (Zimmermann, 1998, p. 11).

Another definition, related to information security, is given by Menezes, van Oorschot, and Vanstone (1996, p. 4) stating that "cryptography is the study of mathematical techniques related to aspects of information security such as confidentiality, data integrity, entity authentication, and data origin authentication". Relating this to privacy, cryptography can be seen as a service used to keep the content of information from all but those authorized to have it.

Information that can be accessed and understood without any special measures is referred to as *plaintext* or *cleartext*. The method by which plaintext is transformed into illegible form is called *encryption*. The resulting

information, after encryption, is referred to as *ciphertext*. Encryption ensures that information remains hidden from all but the intended recipient. When the recipient reverts the ciphertext back to the original plaintext the process is called *decryption*. Figure A.1 illustrates this process.

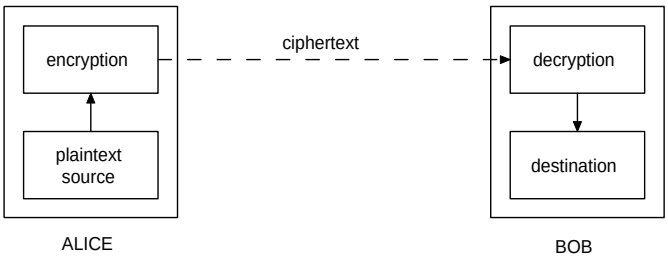

Figure A.1: Encryption And Decryption

In the figure, Alice encrypts some plaintext and transmits the ciphertext to Bob who then decrypts it back into plaintext.

A.2 How Cryptography Works

Cryptography utilizes a *cryptographic algorithm* (or *cipher*) in combination with a *key* to encrypt or decrypt information. Keys consist of very large numerical values, with the secureness of the information directly proportional to the size of the key. Depending on the key the plaintext will encrypt to different ciphertext. The security of the information is reliant on the strength of the cryptographic algorithm and the secrecy of the key.

Encryption works on the basis that the key is only known by the sender and recipient of confidential information. During the encryption process an encryption algorithm is selected which uses the key to create the ciphertext, as illustrated in Figure A.2.

In the example, Alice uses the secret key, the confidential plaintext and the encryption algorithm to create the encrypted ciphertext data.

"In practise, using encryption becomes centered on protecting and managing these secret keys" (Freeman & Jones, 2003, p. 300). One approach requires that both the sender and recipient know and use the same key. This is known as *symmetric* encryption. Another approach is *asymmetric* encryption which utilizes a pair of keys, one for encryption and the other for

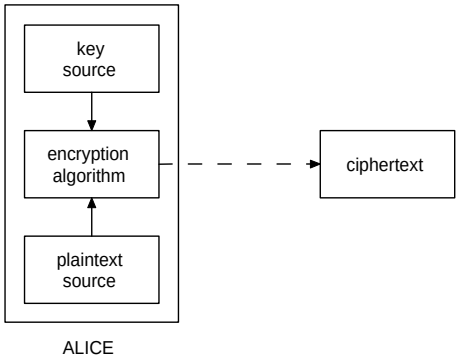

Figure A.2: Encryption Process

decryption. These two methods are examined next.

A.3 Symmetric Encryption

In symmetric encryption, also called *secret-key* encryption, the same key is used both for encryption and decryption. It is the oldest and most established way of ensuring information confidentiality. Figure A.3 below illustrates the symmetric encryption process.

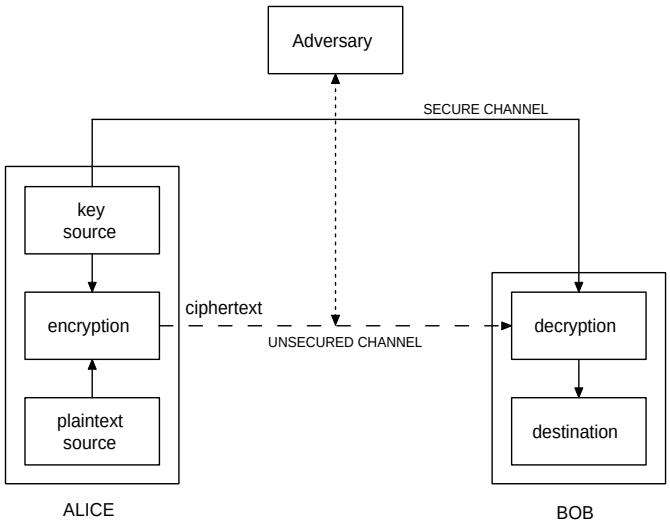

Figure A.3: Symmetric Encryption

In the figure, Alice and Bob agree on a secret key and exchange it in a secure way. Alice then uses the secret key to encrypt the confidential data

and sends it to Bob. Bob decrypts the data using the secret key and reads the message.

The most important issue with symmetric encryption is to find an efficient method to agree upon and exchange keys securely. This problem is referred to as the *key distribution problem*. A possible solution to this problem is through the use of asymmetric encryption.

A.4 Asymmetric Encryption

The concept of asymmetric encryption, or *public key cryptography*, was introduced in 1975 by Whitfield Diffie and Martin Hellman (Zimmermann, 1998, p. 14). It uses a pair of keys: a *public key* for encrypting information and a corresponding *private key* for decryption. You publish your public key to the world while keeping your private key secret. Anyone with a copy of your public key can then encrypt information which only you can read. This scheme is illustrated in Figure A.4.

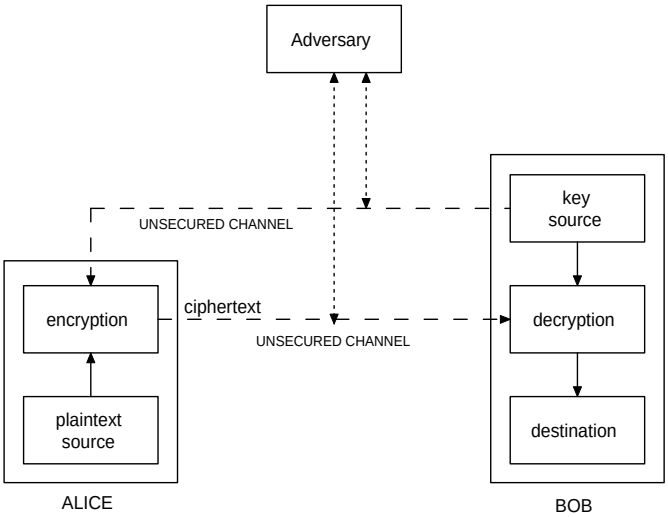

Figure A.4: Asymmetric Encryption

In the example, Bob selects a public and private key. He sends the public key to Alice and keeps the private key secret. Alice uses the public key to encrypt the confidential data and sends it to Bob. Bob decrypts the data using the private key and reads the message.

The main limitation of asymmetric encryption is that it is very slow compared to symmetric encryption. This makes it impractical for large amounts of data. One of the most common applications of asymmetric encryption is to solve the key distribution problem for symmetric encryption. The symmetric key used for encryption can be secured by encrypted it with the recipients public key. The recipient can then retrieve the key by decrypting the message with the corresponding private key. This process is shown in Figure A.5 below.

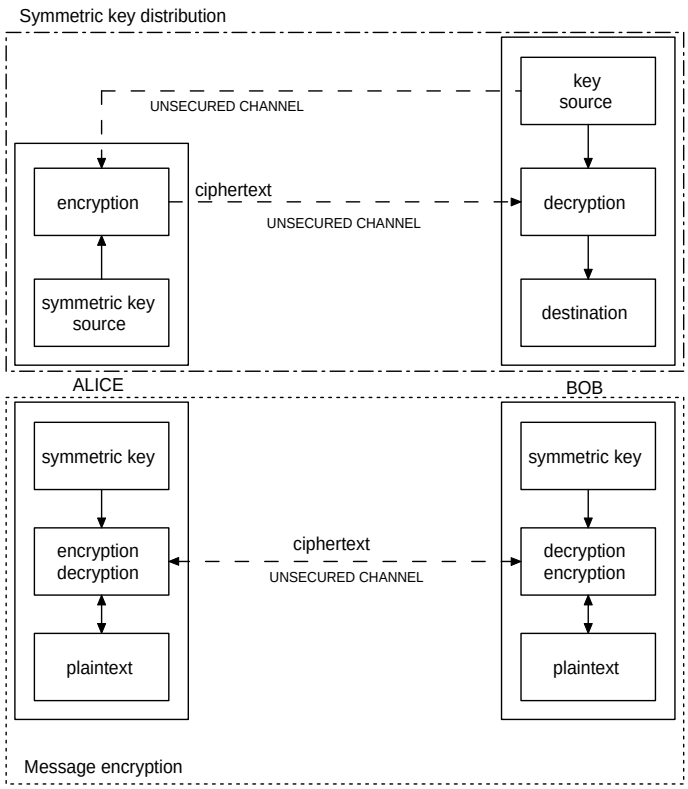

Figure A.5: Symmetric Key Distribution

In the figure, Alice asks Bob for his public key. She then selects a secret key and encrypts it using an asymmetric algorithm with Bob's public key. This is sent to Bob who is the only person able to decrypt the message using his private key and retrieve the secret key. From now on all communications between Alice and Bob can make use of symmetric encryption using the secret key.

Using this scheme the exchange of symmetric keys becomes a simple and secure process, combining the convenience of asymmetric encryption with

the speed of symmetric encryption.

A.5 Cryptography In The .NET Framework

The .NET Framework provides an extensive class library for both symmetric and asymmetric encryption.

Symmetric encryption is provided through several classes that extend the `System.Security.Cryptography.SymmetricAlgorithm` class. These classes provide support for the following encryption algorithms:

- Data Encryption Standard (DES), published in 1976,
- RC2, developed by Ron Rivest and one of the most commonly used symmetrical ciphers,
- Triple-DES, designed to prolong the life of DES and used in the international financial markets for encrypting banking data, and
- Rijndael, also known as the Advanced Encryption Standard (AES), currently the principal U.S. federal standard for data encryption.

Support for asymmetric encryption is provided by classes that extend the `System.Security.Cryptography.AsymmetricAlgorithm` class. The RSA algorithm, created by Ronald Rivest, Adi Shamir, and Leonard Adleman in 1977, is the only one implemented in the .NET Framework. This algorithm is the basis for numerous security systems, and remains the most widely used and understood asymmetric algorithm (Freeman & Jones, 2003, p. 371).

The .NET Framework can also be extended to include different symmetric and asymmetric algorithms as required.

A.6 Conclusion

This appendix provided a high-level overview of cryptographic concepts and terminology. Any application that processes sensitive information should use cryptography. Cryptography, however, can only be effective if implemented correctly. "The most common problem with security software is that complex algorithms are not correctly implemented or are used in such a way as to make them much less secure than they should be ..." (Freeman & Jones, 2003,

p. 305). Only when correctly implemented can cryptography provide an effective technical means to information security.

Appendix B

Academic Paper

An academic paper based on the research contained in this dissertation has been written and is available on the accompanying CD.

The paper entitled, "WSP^3: A Web Service Model for Personal Privacy Protection", will be submitted to a suitable conference.

References

ACM. (1992). *Code of Ethics and Professional Conduct, Section 1.7.* Available from: `http://www.acm.org`. (Last cited: 03 Jun 2002)

Adams, A. (2000). Multimedia information changes the whole privacy ballgame. In *Proceedings of the tenth conference on Computers, Freedom and Privacy* (pp. 25–32). ACM Press.

Bathe, P. (2002). *The Platform for Privacy Preferences Project (P3P).* Available from: `http://www.4guysfromrolla.com`. (Last cited: 14 Nov 2002)

BBBOnline. (2003). *BBBOnline Web site.* Available from: `http://www.bbbonline.org`. (Last cited: 01 Jan 2003)

Bellman, S., Johnson, E. J., & Lohse, G. L. (2001). On site: to opt-in or opt-out?: it depends on the question. *Communications of the ACM*, *44*(2), 25–27.

Belsey, A., & Chadwick, R. (1992). *Ethical Issues in Journalism and the Media.* London: Routledge.

Berners-Lee, T., Masinter, L., & McCahill, M. (1994). *Uniform Resource Locators (URL).* Available from: `http://www.ietf.org`: The Internet Society. (RFC: 1738; Last cited: 25 Jun 2003)

Biron, P. V., & Malhotra, A. (2003). *XML Schema Part 2: Datatypes.* Available from: `http://www.w3.org`: W3C Recommendation. (Last cited: 25 Jun 2003)

Bray, T., Hollander, D., & Layman, A. (1999). *Namespaces in XML.* Available from: `http://www.w3.org`: W3C Recommendation. (Last cited: 25 Jun 2003)

Bray, T., Paoli, J., Sperberg-McQueen, C. M., & Maler, E. (1999). *Extensible Markup Language (XML) 1.0 (Second Edition).* Available from: `http://www.w3.org`: W3C Recommendation. (Last cited: 25 Jun 2003)

Cavoukian, A., & Crompton, M. (2000). *Web Seals: A Review of Online Privacy Programs.* Available from: `http://www.privacy.gov.au`. (Last cited: 02 Jun 2003)

Center for Democracy and Technology (CDT). (2000). *CDT's guide to online privacy.* Available from: `http://www.cdt.org`. (Last cited: 26 May 2002)

Cerami, E. (2002). Web Services Essentials. O'Reilly.

Chappell, D., & Jewell, T. (2002). Java Web Services. O'Reilly.

Christensen, E., Curebera, F., Meredith, G., & Weerawarana, S. (2001). *Web Services Description Language (WSDL) 1.1.* Available from: `http://www.w3.org`: W3C Note. (Last cited: 25 Jun 2003)

Constitution of the Republic of South Africa. (1996). *Constitution of the Republic of South Africa.* Available from: `http://www.polity.org.za`. (Last cited: 12 Aug 2002)

Consumers International. (2001). *Consumer Privacy Threatened On The Net.* Available from: `http://www.consumersinternational.org`. (Last cited: 12 Aug 2002)

Council of Europe (COE). (1981). *Convention for the Protection of Individuals with regard to the Automatic Processing of Personal Data Convention.* Available from: `http://www.coe.fr`. (Last cited: 12 Aug 2002)

Coursey, D. (2002). *Why Web services will be the Next Big Thing.* Available from: `http://www.zdnet.com`. (Last cited: 18 Apr 2002)

Coyle, K. (1999). *P3P: Pretty poor privacy? A social analysis of the Platform for Privacy Preferences.* Available from: `http://www.kcoyle.net`. (Last cited: 22 Oct 2002)

Cranor, L. (2002). The role of privacy advocates and data protection authorities in the design and deployment of the platform for privacy preferences. In *Twelfth Conference on Computers, Freedom and Privacy* (pp. 1–8). Available from: `http://www.cfp2002.org`. (Last cited: 25 Jun 2003)

Cranor, L. F. (2000). Ten years of computers, freedom and privacy: a personal retrospective. In *Proceedings of the tenth conference on Computers, Freedom and Privacy* (pp. 11–15). ACM Press.

Cranor, L. F., Reagle, J., & Ackerman, M. S. (1999). *Beyond Concern: Understanding Net Users' Attitudes About Online Privacy.* Available from: `http://www.research.att.com`. (Last cited: 01 Jun 2002)

Eastlake, D., Reagle, J., Imamura, T., Dillaway, B., & Simon, E. (2002). *XML Encryption Syntax and Processing.* Available from: `http://www.w3.org`: W3C Recommendation. (Last cited: 25 Jun 2003)

Eder, P. (1994). Privacy on parade: Your secrets for sale! *Futurist*, *28*(4), 38–42.

Electronic Communications and Transactions Act. (2002). *Electronic Communications and Transactions Act.* Available from: `http://www.polity.org.za`. (Last cited: 17 Sep 2002)

Electronic Privacy Information Center (EPIC). (1997). *Surfer Beware: Personal Privacy and the Internet.* Available from: `http://www.epic.org`. (Last cited: 22 Oct 2002)

Electronic Privacy Information Center (EPIC). (2000). *Pretty Poor Privacy: An Assessment of P3P and Internet Privacy.* Available from: `http://www.epic.org`. (Last cited: 22 Oct 2002)

EPIC and Privacy International. (2000a). *Privacy And Human Rights 2000 – An International Survey Of Privacy Laws And Developments: Country Reports.* Available from: `http://www.privacyinternational.org`. (Last cited: 12 Aug 2002)

EPIC and Privacy International. (2000b). *Privacy And Human Rights 2000 – An International Survey Of Privacy Laws And Developments:*

Overview: The State of the World's Privacy. Available from: `http://www.privacyinternational.org`. (Last cited: 12 Aug 2002)

EPIC and Privacy International. (2000c). *Privacy And Human Rights 2000 – An International Survey Of Privacy Laws And Developments: Threats to Privacy.* Available from: `http://www.privacyinternational.org`. (Last cited: 12 Aug 2002)

Fallside, D. C. (2003). *XML Schema Part 0: Primer.* Available from: `http://www.w3.org`: W3C Recommendation. (Last cited: 25 Jun 2003)

Federal Trade Commision (FTC). (2000). *Privacy Online: Fair Information Practices in the Electronic Marketplace.* Available from: `http://www.ftc.gov`. (Last cited: 22 Oct 2002)

Fielding, R., Irvine, U., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., & Berners-Lee, T. (1999). *Hypertext Transfer Protocol – HTTP/1.1.* Available from: `http://www.ietf.org`: The Internet Society. (RFC: 2616; Last cited: 25 Jun 2003)

Firefly, Netscape, & Microsft. (1997). *Firefly, Netscape And Microsoft Cooperate To Build Upon Previously Proposed Ops Standard For Personalization With Privacy.* Available from: `http://wp.netscape.com`. (Last cited: 22 Oct 2002)

Fletcher, T., Gradidge, M., & Kingdon, E. (2002). *Commentary on the Electronic Communications and Transactions Bill, 2002.* Available from: `http://www.cliffedekker.co.za`. (Last cited: 17 Sep 2002)

Freedman, W. (1987). *The Right of Privacy in the Computer Age.* New York: Quorum Books.

Freeman, A., & Jones, A. (2003). *Programming .NET Security.* O'Reilly.

Glass, G. (2000). *The Web services (r)evolution.* Available from: `http://www-4.ibm.com`. (Last cited: 18 Apr 2002)

Grimm, R., & Rossnagel, A. (2000). Can P3P help to protect privacy worldwide? In *Proceedings of the 2000 ACM workshops on Multimedia* (pp. 157–160). ACM Press.

Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J.-J., & Nielsen, H. F. (2003a). *SOAP Version 1.2 Part 0: Primer.* Available from: `http://www.w3.org`: W3C Recommendation. (Last cited: 25 Jun 2003)

Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J.-J., & Nielsen, H. F. (2003b). *SOAP Version 1.2 Part 1: Messaging Framework.* Available from: `http://www.w3.org`: W3C Recommendation. (Last cited: 25 Jun 2003)

Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J.-J., & Nielsen, H. F. (2003c). *SOAP Version 1.2 Part 2: Adjuncts.* Available from: `http://www.w3.org`: W3C Recommendation. (Last cited: 25 Jun 2003)

Hallett, T. (2003). *Web services "ready for prime time".* Available from: `http://www.silicon.com`. (Last cited: 02 Sep 2003)

Henderson, S. C., & Snyder, C. A. (1999). Personal information privacy: implications for MIS managers. *Information & Management*, *36*, 213–220.

Hensley, P., Metral, M., Shardanand, U., Converse, D., & Myers, M. (1997). *Implementation of OPS Over HTTP.* Available from: `http://www.w3.org`: W3C Note. (Last cited: 25 Jun 2003)

Hinde, S. (2001). 2001: A Privacy Odyssey. *Computers and Security*, *20*(1), 21–27.

Hinde, S. (2002). 2001: A Privacy Odyssey Revisited. *Computers and Security*, *21*(1), 16–34.

Hochheiser, H. (2002). The Platform for Privacy Preference as a Social Protocol: An Examination Within the U.S. Policy Context. *ACM Transactions on Internet Technology*, *2*(4), 276–306.

Hyman, G. (2002). *Industry Majors Make the Push for Web Services.* Available from: `http://siliconvalley.internet.com`. (Last cited: 15 Apr 2002)

Kirtland, M. (2001a). *A Platform for Web Services.* Available from: `http://msdn.microsoft.com`. (Last cited: 15 Jun 2002)

Kirtland, M. (2001b). *User Privacy.* Available from: http://msdn.microsoft.com. (Last cited: 15 Jun 2002)

Klensin, J. (2001). *Simple Mail Transfer Protocol.* Available from: http://www.ietf.org: The Internet Society. (RFC: 2821; Last cited: 25 Jun 2003)

Kristol, D., & Montulli, L. (2000). *HTTP State Management Mechanism.* Available from: http://www.ietf.org: The Internet Society. (RFC: 2965)

Kristol, D. M. (2001). HTTP Cookies: Standards, Privacy, and Politics. *ACM Transactions on Internet Technology*, *1*(2), 151–198.

Livingston, D. (2002). Advanced SOAP for Web Development. Prentice Hall Computer Books.

Madsen, P., & Adams, C. (2002). *Privacy and XML, Part I.* Available from: http://www.xml.com. (Last cited: 22 Apr 2002)

Magid, L. J. (1997). *Personal privacy : Steps to protect you, the consumer.* Available from: http://www.larrysworld.com. (Last cited: 22 Oct 2002)

Marchiori, M. (2002). *The Platform for Privacy Preferences 1.0 (P3P1.0) Specification.* Available from: http://www.w3.org: W3C Recommendation. (Last cited: 22 Oct 2002)

Mason, R. (1986). Four ethical issues of the information age. *MIS Quarterly*, *10*(1), 4–12.

Media Awareness Network. (1999). *International Privacy and Freedom of Information Laws.* Available from: http://www.media-awareness.ca. (Last cited: 12 Aug 2002)

Menezes, A., van Oorschot, P., & Vanstone, S. (1996). *Handbook of Applied Cryptography.* CRC Press. (For further information see: http://www.cacr.math.uwaterloo.ca)

Moats, R. (1997). *URN Syntax.* Available from: http://www.ietf.org: The Internet Society. (RFC: 2141; Last cited: 25 Jun 2003)

Newcomer, E. (2003). Understanding Web Services: XML, WSDL, SOAP, and UDDI. Addison-Wesley.

OASIS. (2003). *Universal Description, Discovery, and Integration.* Available from: `http://www.uddi.org`: OASIS. (Last cited: 25 Jun 2003)

Oberholzer, H. (2001). *A Privacy Protection Model to Support Personal Privacy in Relational Databases.* Unpublished master's thesis, Rand Afrikaans University.

Organization for Economic Cooperation and Development (OECD). (1981). *Guidelines Governing the Protection of Privacy and Transborder Data Flows of Personal Data.* Available from: `http://www.oecd.org`. (Last cited: 12 Aug 2002)

Peng, S. L. (2002). *Figuring Out Web Services.* Available from: `http://www.internetnews.com`. (Last cited: 22 Apr 2002)

Privacilla. (2001a). *Privacy Fundamentals.* Available from: `http://www.privacilla.org`. (Last cited: 26 May 2002)

Privacilla. (2001b). *What is Privacy?* Available from: `http://www.privacilla.org`. (Last cited: 26 May 2002)

Raggett, D., Le Hors, A., & Jacobs, I. (1999). *HTML 4.01 Specification.* Available from: `http://www.w3.org`: W3C Recommendation. (Last cited: 25 Jun 2003)

Reagle, J., & Cranor, L. F. (1999). The platform for privacy preferences. *Communications of the ACM*, *42*(2), 48–55.

Regulation of Interception (and Monitoring) of Communications Act. (2001). *Regulation of Interception (and Monitoring) of Communications Act.* Available from: `http://www.privacy.org`. (Last cited: 12 Aug 2002)

Robertson, C. J., & Sarathy, R. (2002). Digital privacy. *Business Horizons.*

Roy, J., & Ramanujan, A. (2001). Understanding Web Services. *IT Pro*, *3*(6), 69–73.

Shohoud, Y. (2003). Real World XML Web Services. Pearson Education.

Short, S. (2002). *Building XML Web Services for the Microsoft .NET Platform.* Redmond: Microsoft Press.

Skonnard, A. (2001). Understanding XML Namespaces. *MSDN Magazine.* (Available from: `http://msdn.microsoft.com`)

Skonnard, A. (2003). *Understanding SOAP.* Available from: `http://msdn.microsoft.com`. (Last cited: 24 Mar 2003)

Smith, H. (1993). Privacy policies and practices: Inside the organizational maze. *Communications of the ACM*, *36*(12), 105–122.

Smith, J. H. (1993). Privacy policies and practices: inside the organizational maze. *Communications of the ACM*, *36*(12), 104–122.

South African Law Commssion. (2003). *Privacy and Data Protection.* Available from: `http://www.law.wits.ac.za/salc`. (Last cited: 27 Aug 2003)

Srinivasan, R. (1995). *RPC: Remote Procedure Call Protocol Specification Version 2.* Available from: `http://www.ietf.org`: The Internet Society. (RFC: 1831; Last cited: 25 Jun 2003)

Swire, P. P. (2000). Towards Digital eQuality: the U.S. Government working group on electronic commerce. In *Proceedings of the tenth conference on Computers, Freedom and Privacy* (pp. 281–284). ACM Press.

Tapang, C. (2001). Web Services Description Language (WSDL) Explained. *MSDN Magazine.* (Available from: `http://msdn.microsoft.com`)

Tavani, H. T. (1999). Privacy online. *ACM SIGCAS Computers and Society*, *29*(4), 11–19.

TechTarget. (2001). *Privacy - A Whatis Definition.* Available from: `http://www.techtarget.com`. (Last cited: 26 May 2002)

The Privacy Committee of New South Wales. (2002). *International Law and Legislation.* Available from: `http://www.austlii.edu.au`. (Last cited: 12 Aug 2002)

The Privacy Hub. (2002). *Privacy Definition.* Available from: `http://dsa-isis.jrc.it`. (Last cited: 26 May 2002)

Thompson, H. S., Beech, D., Maloney, M., & Mendelsohn, N. (2001). *XML Schema Part 1: Structures.* Available from: `http://www.w3.org`: W3C Recommendation. (Last cited: 25 Jun 2003)

TRUSTe. (2003). *TRUSTe Web site.* Available from: `http://www.truste.org`. (Last cited: 01 Jan 2003)

Universal Declaration of Human Rights. (1948). *Universal Declaration of Human Rights.* Available from: `http://www.hrweb.org`. (Last cited: 12 Aug 2002)

Valentine, C., Dykes, L., & Tittel, E. (2001). XML Schemas. SYBEX.

Vaughan-Nichols, S. J. (2001). *The Woes of Web Services.* Available from: `http://www.byte.com`. (Last cited: 18 Apr 2002)

WebTrust. (2003). *WebTrust Web site.* Available from: `http://www.webtrust.net`. (Last cited: 01 Jan 2003)

Weckert, J., & Adeney, D. (1997). *Computer And Information Ethics.* Westport: Greenwood Press.

Zimmermann, P. (1998). *An Introduction to Cryptography.* Network Associates, Inc.