Introducing Hippocratic Log Files for Personal Privacy Control

by

Andrew Rutherford

Introducing Hippocratic Log Files for Personal Privacy Control

by

Andrew Rutherford

Dissertation

submitted in fulfillment
of the requirements
for the degree

Magister Technologiae

in

Information Technology

in the

Faculty of Engineering

of the

Nelson Mandela Metropolitan University

Promoter: Prof. Reinhardt A. Botha

January 2005

Declaration

I, Andrew Rutherford, 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 education institute.

Andrew Rutherford

Abstract

The rapid growth of the Internet has served to intensify existing privacy concerns of the individual, to the point that privacy is the number one concern amongst Internet users today. Tools exist that can provide users with a choice of anonymity or pseudonymity. However, many Web transactions require the release of personally identifying information, thus rendering such tools infeasible in many instances. Since it is then a given that users are often required to release personal information, which could be recorded, it follows that they require a greater degree of control over the information they release.

Hippocratic databases, designed by Agrawal, Kiernan, Srikant, and Xu (2002), aim to give users greater control over information stored in a database. Their design was inspired by the medical Hippocratic oath, and makes data privacy protection a fundamental responsibility of the database itself. To achieve the privacy of data, Hippocratic databases are governed by 10 key privacy principles.

This dissertation argues, that asides from a few challenges, the 10 principles of Hippocratic databases can be applied to log files. This argument is supported by presenting a high-level functional view of a Hippocratic log file architecture. This architecture focuses on issues that highlight the control users gain over their personal information that is collected in log files. By presenting a layered view of the aforementioned architecture, it was, furthermore, possible to provide greater insight into the major processes that would be at work in a Hippocratic log file implementation. An exploratory prototype served to understand and demonstrate certain of the architectural components of Hippocratic log files. This dissertation, thus, makes a contribution to the ideal of providing users with greater control over their personal information, by proposing the use of Hippocratic log files.

Acknowledgements

God the Father, who knows all our secrets but never violates our privacy.

My *loving wife Aletta*, without whose unwavering support, this work would not have been possible.

My promoter, *Prof. Reinhardt Botha*, for his motivation and guidance throughout. His willingness to give up so much of his time, even during a holiday period, is very much appreciated.

Prof. Martin Olivier, who planted the seed leading to the investigation into Hippocratic log files and greatly assisted in the publishing of two academic papers.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Information, to coin a cliche, is power. The World Wide Web (WWW) is fast becoming the central location for goods, services and information. It would thus seem fair to extrapolate that the WWW is a prime source of power. But as is often the case, where there is power, there is the potential for the abuse of power.

The WWW has become such a powerful information medium due to its unregulated nature, the high degree of browser and protocol flexibility and the fact that it has evolved into the worlds largest shop, library and chat-room (Froomkin, 2000). These self same factors combine to make the WWW a treasure trove of personal information regarding individual Web users. Rapid technological advancements have made the collection, processing, interpretation and dissemination of personal information increasingly more rapid and feasible (Lin & Loui, 1998; Tavani, 1999). Not surprisingly Internet users have rated loss of privacy as their number one concern when transacting on the Web (Tavani, 1999).

Information can be collected with user consent in a direct manner, for example fill-in-forms, and indirectly and without user consent, for example cookies and log files. It is the indirect manner of collection that raises the most user concerns and objections. As user reliance on the Internet grows, so to will their privacy concerns. This reliance will increase the likelihood that data regarding their interests, preferences and economic behaviour will be recorded (Froomkin, 2000). This information can be used to create profiles of individual users. While such profiles can be used for valid purposes e.g. Web site personalization and customization, they can also be used for more

subversive reasons e.g. denying access to services. Internet users become particularly incensed when data is collected without consent, or used for a purpose other than that for which it was collected (Rose, 2001).

Addressing the privacy concerns raised by information collected in Web server log files is the primary motivation for this study.

1.1 Motivation for this Study

Privacy and the loss thereof is the number one concern of Internet users today (Tavani, 1999; Wang, Lee, & Wang, 1998). The Gartner group's view is that privacy concerns will be the greatest inhibitor of consumer-based e-business through 2006 (Rezgul, Bouguettaya, & Eltoweissy, 2003). It is essential to address user concerns, if only to combat the tremendous loss of potential revenue. It is estimated that user privacy concerns in 2002 resulted a loss of $18 billion (Rezgul et al., 2003).

In their efforts to protect their privacy, users may opt out of a service entirely, or provide erroneous data (Rose, 2001). The easiest way to control privacy would not be to divulge any personal information in the first place. To this end, technologies exist that enable users to maintain a degree of anonymity on the Web. However, *anonymity is not always possible*. Many transactions require the release of personal information, for example a credit card number to complete an online purchase (Tavani, 1999; Olivier, 2003). This being said, once personal information is released, it can be recorded, whereupon the information donor may lose a significant degree of control over it (Froomkin, 2000). The solution to this dilemma has to include the continued protection of private information after it has been recorded. Thereby allowing the donors of information to exercise control over their private information after releasing it. Indeed the Computing Research Association (2003) has identified, as one of their four grand challenges, that the computing environments of the future must strive to *"give end-users security they can understand and privacy they can control"*.

To help achieve the protection of private information stored in databases Agrawal et al. (2002) outlined their concept of Hippocratic Databases. Inspired by the medical Hippocratic oath, the primary goal of such databases is the privacy of data they manage. In achieving this goal Hippocratic data-

bases are governed by 10 key principles. Information donors to Hippocratic databases, are afforded the opportunity to provide or deny consent to stated collection purposes.

While Agrawal et al. (2002) focused their attention on the protection of privacy of information held in databases, they encourage *the application of Hippocratic principles to other data sources.* Log files, particularly Web server log files, provide just such an alternate data source for investigation. Users are often unaware that information logging is occurring and even if aware, are ignorant of what information is being recorded (Lin & Loui, 1998). The manner in which log files *unobtrusively collect data* makes it highly appropriate to place them under the Hippocratic spotlight.

1.2 Problem Statement

Concerns over privacy are by no means a recent phenomenon. However, the dawn of the Internet and its rapid growth has fuelled and intensified existing concerns. Internet users often release information without their knowledge or consent. For example, when visiting a Web site, user actions are recorded in a log file. How then can users gain a degree of control over their private information contained in log files, after it has been collected? The concept of Hippocratic databases demonstrates how such control can be given to users with respect to information collected in databases. This then leads us to the fundamental question investigated in this dissertation:

- *"Can the privacy principles of Hippocratic databases be applied to log files?"*.

This question prompts the investigation of a number of issues, some to gain a better understanding of the domain of discourse. Gaining this understanding can be achieved by answering questions along the following lines:

- How can the goal of giving users greater control over their private information be realized?
- Given that anonymity on the Web is not always possible, by what other means can user privacy be assured?

- What impact will the application of Hippocratic principles have on the unobtrusive collection of information by log files?

Answering each of these questions will be the will enable the achievement of the dissertation objectives.

1.3 Objectives

The primary objective of this dissertation is to make a contribution to providing users with greater control over their personal information after it has been collected, specifically with respect to information stored in server log files. This contribution will primarily be achieved by determining whether the privacy principles of Hippocratic databases can be applied to log files. In order to make this determination the following sub-objectives need to be addressed:

- Map the privacy principles of Hippocratic database onto log files and adapt them where necessary and possible.
- Describe a high-level architectural view of a Hippocratic log file architecture, paying particular attention to how users can control personally identifiable information collected unobtrusively.
- Discuss some of the major processes involved in a Hippocratic log file implementation, by delving deeper into the aforementioned architecture.
- Show that the active implementation of Hippocratic log files would bring an end to unobtrusive information logging.

In order to fulfill the primary and sub-objectives, it will of course be necessary to discuss the concept of privacy, as well as current mechanisms available for privacy protection.

1.4 Methodology

The methodology undertaken will include a thorough literature study. This study will encompass the concept of privacy, growing privacy concerns and

privacy enhancing technologies. Log files, particularly their impact on user privacy, will also be studied. The focus will then shift to the investigation of the applicability of Hippocratic database principles to log files. This will be conducted as follows:

- An in-depth discussion of Hippocratic databases, with emphasis on its principles, architecture and challenges.
- The proposal of a high-level architecture highlighting a functional view of a Hippocratic log file implementation. This architecture will pay close attention to how users can gain greater control over information collected unobtrusively in log files.
- A layered view of the aforementioned architecture will provide greater insight into the major processes that will play a role in a Hippocratic log file implementation.
- The development of an exploratory prototype will serve to aid understanding and practically demonstrate certain of the architectural components of Hippocratic log files.

By applying the outlined methodology, the objectives of this dissertation were met.

1.5 Layout of Dissertation

The layout of the dissertation is depicted in figure 1.1. **Chapter 1** will provide some background to the study in order to clearly define the problem domain. **Chapter 2** will present a comprehensive view of privacy and current privacy enhancing technologies. **Chapter 3** will focus on log files and some of the reasons for which information logging occurs. It will also highlight the relation between log files and privacy issues. **Chapter 4** will introduce the concept of Hippocratic databases. It will present the Hippocratic database principles as well as highlight some issues and concerns with regards to the implementation of Hippocratic databases. Initially, **chapter 5** will define the concept of Hippocratic log files, in terms of the Hippocratic database principles introduced in chapter 4. Continuing with the discussion of Hippocratic log files, **chapter 6** presents an architectural view of the concept.

Chapter 7 reports on some experimental investigation that was conducted regarding some of the architectural concepts. **Chapter 8** will conclude the dissertation.

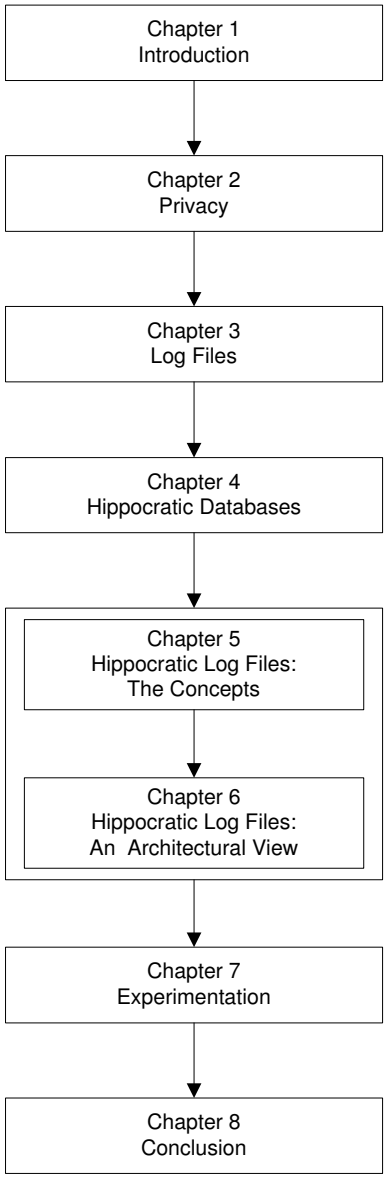


Figure 1.1: Layout of Dissertation

Chapter 2

Privacy

"The fantastic advances in the field of electronic communication constitute a greater danger to the privacy of the individual. " - Earl Warren[1]

"PRIVACY is not something that I'm merely entitled to, it's an absolute prerequisite." - Marlon Brando[2]

There are two things that one can conclusively say regarding privacy. Firstly, that privacy concerns of individuals are on the rise, and secondly, that the notion of privacy is hard to understand and not easily defined (Tavani, 1999). The explosive rate of the Internet and the resultant information explosion has only served to exacerbate the existing privacy dilemma.

This chapter will consider the notion of privacy and provide an overview of the multitude of definitions and understandings of the term. The privacy concerns of users, with particular reference to the Internet, will be discussed. This will be followed by an overview of current initiatives and technologies for maintaining a degree of privacy on the Web.

2.1 What is Privacy?

When studying the privacy literature, it soon becomes clear that privacy is difficult to understand and even more difficult to define (Tavani, 1999). The variety of understandings of privacy serve to prove this point.

In an early definition, Warren and Brandeis (1890) state that privacy is the right to be left alone. The current author agrees that one dimension

[1] American supreme court justice (1891 - 1974)

[2] American Actor, Director

of privacy may very well be the right to be left alone. However, he further believes that this definition as it stands, is too broad. It covers situations that may have very little to do with privacy. For example, an employee that is continuously interrupted whilst trying to finish an important report is not losing privacy, but he may very well not being left alone.

Lin and Loui (1998) cite Alan Westin as stating that privacy is the control of personal information. However, this definition fails to consider the distinction between loss of privacy and a violation of privacy. There may indeed be situations where there is a loss of control over personal information, but where privacy is not violated. For example, I may surrender information to my attorney during litigation proceedings. I as client, may have recourse should my attorney improperly disclose this information, but a certain degree of control is lost on surrendering the information. The relationship between clients and their attorneys is such that even though a degree of control over information is lost on its surrender, there is no violation of privacy.

According to Clarke (1999), "Information privacy refers to the claims of individuals that data about themselves should generally not be available to other individuals and organizations, and that where data is possessed by another party, the individual must be able to exercise a substantial degree of control over that data and its use." Although the current author is reasonably satisfied with such a definition, he is concerned about the implication of the phrase "generally not available". Although it may not be the intention of any individual surfing the Web, the very act of surfing makes information generally available. Web sites routinely log information without the knowledge or consent of users (Lin & Loui, 1998). The individual may thus, after all not have "a substantial degree of control over that data and its use."

There is a school of thought, by some government and corporate executives, that the public is not to be trusted. They believe that individuals only deserve the benefits of modern society if they provide greater access to personal information (Clarke, 1999). To this the author can only say that the argument is killed by its own contradiction. If the public is not to be trusted, why should corporate executives or politicians be considered any more trustworthy in ensuring the ethical and moral use of collected information?

Tavani and Moor (2001) argue that privacy, and control of private information, are two separate, yet related, concepts. They state that privacy

is primarily about protection from intrusion and information gathering by others. Control of information, on the other hand, is a justification of the right to privacy and plays a role in the management of privacy. This notion of privacy is closest to the author's own understanding. Where possible, an individual's information should remain private and uncollected. However, this is not always possible and once information has been collected, it should be managed in such a manner as to ensure no violations of privacy.

Regardless of the definitional aspects of privacy, there seems to be reasonable consensus that privacy is an important part of our lives and deserves protection. The importance attached to privacy by individuals is discussed in the next section.

2.2 Privacy Concerns

Concerns over privacy, and the loss thereof, are certainly not new. Social and technological advancements have more often then not required of man to sacrifice a certain amount of privacy in order to derive full benefit from the advancements. Humanity's move from a hunter-gatherer to agricultural communities came about due to the benefits he derived from the transition; namely increased security, better housing and improved food and water supply (Kaufman, Edlund, Ford, & Powers, 2002). Similar losses of, and concerns for privacy have occurred in more recent times with the invention of the camera and telephone (Tavani, 1999). Technological advancements are, however, occurring at an ever faster rate.

Kaufman et al. (2002) argue that past advancements took place at a rate that allowed "social contracts" to develop in pace with technology. They define a social contract as "the collective rules that constrain the behaviour of individuals and groups living in a society in such a way as to protect the individual while also benefiting society as a whole". The telephone was invented in 1876 but it was not until the 1960's that the number of households in the USA with telephones exceeded 80%. In other words, society had ample time to integrate the telephone into every day life in a manner that was acceptable to all. By comparison, the Internet has rapidly mushroomed into a global network with millions of users online. The Internet became a part of daily life without the societal norms for its ethical and just use having been

firmly established. It is thus hardly surprising that privacy is said to be the number one issue facing the Internet (Tavani, 1999).

However, privacy was a major concern prior to the advent of the Internet. The question then becomes, "why exactly has the Internet heightened user privacy concerns?" The reasons for this, according to Tavani (1999) and Treese (2000), are twofold. The Internet has (a) greatly enhanced existing privacy threats while (b) bringing about new and unique threats of its own. The next 2 subsections will address these issues in greater detail.

2.2.1 Internet Enhanced Threats

Tavani (1999) broadly categorizes the enhanced threats into data collection and the subsequent use of collected data. He is supported by Clarke (1999) who states, "Data is increasingly collected and personalized. Storage technology ensures it remains available. Database technologies make it discoverable. And telecommunications enables its rapid reticulation." Certainly the Internet has greatly increased the ability to monitor and record user activity, and indeed has enabled new kinds of information to be recorded (Lin & Loui, 1998).

This recording of information takes place in both direct and indirect manners. Directly, for example, by means of fill-in-forms where a user is openly requested for information. The use of server log files is an example of an indirect method of information collection. Here the user surrenders information, in all likelihood without their knowledge or consent.

The combination of directly and indirectly collected information is becoming a common occurrence, particularly for the purposes of targeted advertising. Thus, while the collection and monitoring of individuals is not new, the Internet has dramatically increased the scale at which these activities can be performed (Tavani, 1999).

The sheer volume of personal information that can be collected, has made it a marketable commodity. The sale of collected information to third parties can be profitable. Unfortunately, this sale often takes place without the knowledge or consent of the individual (Tavani, 1999). The wealth of information collected via the Internet also makes it a particularly attractive source for data mining activities, thus allowing the discovery of new information and relationships in the data (Tavani, 1999; Clarke, 1999).

2.2.2 Internet Specific Threats

The Internet has also brought with it a host of threats to individual privacy. A few of these threats will be highlighted and discussed in this section.

The Hypertext transfer protocol (HTTP) provides the rules and conventions that enable web sites and browsers to communicate. HTTP is a stateless protocol. This stateless nature is best explained by describing what transpires when a user clicks on a hyperlink or types in a Web address in their browser (Kristol, 2001):

- A user clicks on hyperlink, making a request of a server.
- The user agent (browser) connects to the Web server sending it a request for the desired information.
- The Web server responds by returning the desired information.
- On receipt of the server response the browser disconnects from the server.

The process is stateless since a user must establish a new connection to the server each time they have a request. Even though a user may direct multiple requests to a server during a browsing session, each is treated as if it was the first.

The quest to overcome this statelessness, lead to Netscape creating *Internet Cookies* in the mid-1990s (Berghel, 2001). Cookies are stored as text files on a users machine. Sit and Fu (2001) define cookies as "a key/value pair sent to a browser by a Web server to capture the current state of a Web session." Once a cookie has been created, it is automatically returned to the creating server on subsequent requests hereby allowing state management.

Cookies can of course be very beneficial to users. They may store passwords which would otherwise be forgotten, or store user preferences allowing Web pages to be customized to individual tastes (Froomkin, 2000). One of the most common uses for cookies is to store a session identifier. This allows a site to connect a set of related requests from the same browser (Treese, 2000). Cookies can collect non-identifiable information without user consent (Lin & Loui, 1998). Whilst the morality of this may be questionable, it poses few privacy problems.

The privacy problem becomes truly apparent when one considers the implication of truly identifiable information, e.g. ID numbers and addresses, being stored in a cookie. A site hosting health information would be able to build a list of persons searching for information on AIDS. Such information could be sold to third parties resulting in individuals being subjected to price and other forms of discrimination. Of course, for personally identifiable information to be stored in a cookie, a user would have to freely provide it. However, they may do so for one reason, without the knowledge, or indeed the consent, that (a) it will be stored for perpetual use, and (b) that it be used for a purpose other then originally intended (Lin & Loui, 1998; Berghel, 2001).

A user may even receive cookies from 'unvisited' sites. Such cookies have been termed *third-party cookies* due to the fact that they are sent from a site which the user is not actively viewing (Berghel, 2002). This is possible primarily because a Web page is not one entity, but is rather made up of individual parts combining to create a whole. These individual parts may be located on multiple, geographically dispersed servers. In this manner, Doubleclick, an Internet advertising agency, provides banner advertisements for many Internet sites. Through the use of cookies Doubleclick is able to track the movement of Internet users between Doubleclick affiliated sites. Perhaps even more disturbing is that they can also track how often users visit these sites, as well as what they view while they are there (Froomkin, 2000).

An even more invasive privacy threat, is that of the so-called *Web bug*. This is a spin off of the original cookie concept. A Web bug is a graphic, represented as an HTML <img> tag on a Web page or in an e-mail message. It is designed for the specific purpose of monitoring who is reading the page or message. They are often invisible as they are typically only 1-by-1 pixel in size (Martin, 2003). They are normally placed on Web pages by companies not affiliated with the hosting site, and are used as a means to track Internet users surfing habits, without their knowledge and consent. If many different pages from different sites use the same Web bug, the bug creator can build up a lot of information regarding where a user has been. Depending on what other information is shared with that company, it may be able to link that to personally identifiable information e.g. name or email address (Treese,

2000).

Downloading shareware and freeware software from the Internet also poses a huge privacy threat. *Spyware* is software that secretly collects user information typically without user knowledge and consent. The collected information is then returned to the software operator using the users' own internet connection (Webopedia, 2004; Surferbeware.com, 2003; McManus, 2002). Spyware often exists as a hidden component of freeware or shareware programs downloadable from the Internet. Thus, they are unknowingly installed along with the downloaded program (Webopedia, 2004; McManus, 2002). Many freeware and shareware authors are under the impression that since they are providing their software for free or at a reduced price they have the right to profit from users personal information. Spyware authors either use the collected information for marketing or advertising purposes; or alternatively they may sell the information to a third party (Webopedia, 2004).

Even *search engine* facilities, used to locate information on the Internet, may pose a threat to privacy. Internet search engines make it very easy to discover information about individuals because if your name appears anywhere on the internet, chances are a search engine can find it (Tavani, 1999). While writing this dissertation this author did a search for his own name and surname. Within a very short space of time the following information was found:

- His place of work, email address and office telephone number;
- that he was a module tutor for Database theory for a distance education program;
- that he presented a paper at a computer symposium in South Africa in 2003;
- that he belongs to the Secure Research Workflow Group chaired by Professor Reinhardt Botha; and
- that he is listed as a reference on a past student's curriculum vitae.

All of this information was discovered within 10 minutes of having initiated the search request.

It is clear from the aforegoing that privacy concerns predate the Internet. But that the Internet has indeed become the focal point of peoples' privacy fears is equally indisputable. However, Internet users often interact with Web sites that neither require nor benefit from the receipt of personal information. The Internet even makes it possible for much information to be gathered without user knowledge or consent. In an attempt to combat this, various tools and technologies have been developed to aid Internet users in their quest for privacy protection. These tools and technologies can be grouped together according to the manner in which they aim to protect privacy. These groups will be discussed in greater details in the sections that follow.

2.3 Privacy Through Anonymity

Tools providing anonymity allow individuals to transact on the Internet without disclosing any personally identifiable information (Rao & Rohatgi, 2000). The key characteristic of an anonymous transaction is that the identity of one or more transacting parties cannot be determined from the data itself, nor by combining the transaction with other data (Clarke, 1996). A few tools which enable anonymous transactions shall now be discussed.

2.3.1 Anonymizing Proxies

Internet users may subscribe to an anonymizing proxy service. In such a scenario, all of a user's HTTP requests are routed to the proxy-based anonymizer, before submission to the destination site. Since all requests are submitted by the proxy, the only IP address revealed to visited Web sites is that of the proxy (Cranor, 1999). Figure 2.1 depicts such interaction. A pitfall of anonymizing proxies, is that users will need to place a great deal of trust in them. While their requests will be anonymous to the destination site, they will not be anonymous to the proxy itself. Additionally, a user's own Internet service provider (this may be their employer) may log their Web activities (Cranor, 1999; Treese, 2000).

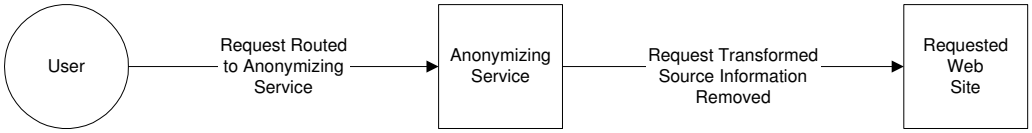


Figure 2.1: Proxy-based Anonymizing

2.3.2 Crowds

The crowds concept is based on the idea that "people can be anonymous when they blend into a crowd" (Cranor, 1998). Geographically dispersed users are collected into a group called a "crowd", that performs Web transactions on behalf of its members (Reiter & Rubin, 1999). Users run a process on their local machine called a "jondo" which gives them access to the crowd. Joining the crowd merely requires the starting of this jondo on the local machine. Having started, the jondo announces it's presence to the other crowd members and is itself informed of the current crowd members (Reiter & Rubin, 1999). A user's jondo uses the Crowd for Web navigation and in so doing maintain anonymity. The basic operation for the first user request after joining a crowd, is as follows (Reiter & Rubin, 1999; Cranor, 1998):

1. The User makes an HTTP request.
2. The Request is sent to his jondo.
3. The jondo randomly selects another member of the crowd and forwards the request to this member.
4. The receiving crowd members jondo makes a random choice to either (a) forward the request to its destination; or (b) to forward the request to another randomly selected crowd member.
5. The process continues until the message is received by the originally requested server.

Figure 2.2, graphically depicts example paths in a crowd. The request initiator and destination server, of each path, have been assigned the same name to facilitate ease of understanding.

The sequence of jondos a request follows to reach its final destination is called a "path" and each individual communication between jondos along the

path is called a "hop". Once the receiving server has received the request, it replies to the original user by sending its response backward along the path. Once a path has been established for a crowd member, it remains static for all of its subsequent Web interaction. This is achieved by each path being assigned a unique identifier. Each jondo on the path keeps track of its predecessor and successor for that path (Reiter & Rubin, 1999). The path will only change in the event of a jondo failing or new jondos joining the crowd. This is best understood by viewing figure 2.2. If one follows the request path of user 1, it is clear that there is only one hop in its path prior to reaching its destination. A request by user 5 on the other hand has two such hops.

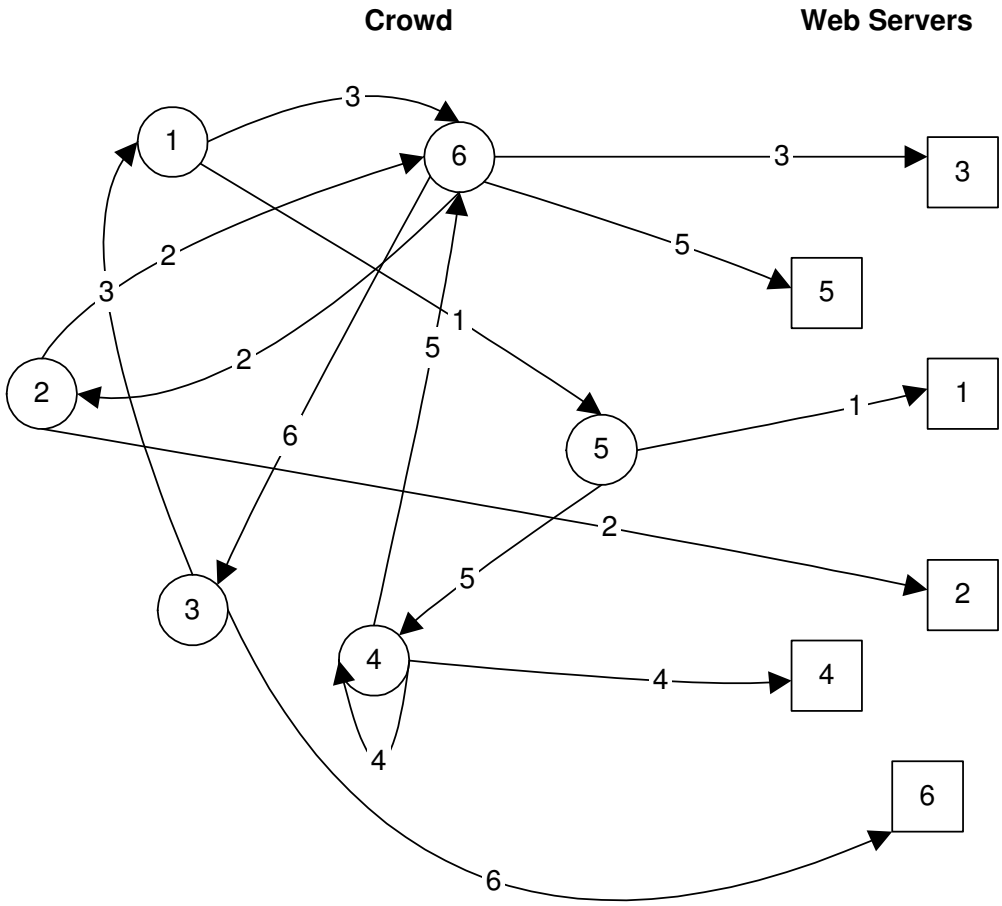


Figure 2.2: Paths in a Crowd (Reiter & Rubin, 1999)

Thus, the manner in which Web traffic is managed within the crowd establishes communication anonymity. By the time a user request reaches its

final destination it is impossible for the destination server, or for that matter any of the other crowd members, to determine which member initiated the request (Reiter & Rubin, 1999; Cranor, 1998).

Crowd's major advantage over anonymizing proxies is that there is no single point at which all users' anonymity can be lost – as stated earlier, users have to trust their anonymizing proxy to maintain their anonymity. A potential drawback to crowds is that request content is not hidden from jondos along the path. This could be problematic when request contents contain such information as username and password. However, as Reiter and Rubin (1999) point out, such communication will expose user identity to the destination server anyway, thus rendering anonymizing counterproductive. Such communications would do better to disable crowds and communicate directly with the destination server. Additionally, from a performance standpoint, retrieval times will be lengthened in the crowds environment due to the path traversal required to fulfill a user request.

2.3.3 Onion Routing

Onion routing is based on the notion of mixing the connections from different users, making it difficult to determine who is communicating with whom (Rezgul et al., 2003). This technique achieves its goal by dynamically building anonymous connections within a network of real-time Chaum mixes (Goldschlag, Reed, & Syverson, 1999). A Chaum mix is a store and forward device which accepts fixed-length messages from numerous sources. It then subjects the messages to cryptographic transformations and forwards them to the next destination in random order (Rezgul et al., 2003). Using multiple networked mixes makes message tracking extremely difficult. The goal of onion routing is two-fold. Firstly, to protect the privacy of the sender and recipient of a message, and secondly to protect message content as it traverses a network of Onion Routers. The routing onion concept is the main innovation of onion routing. This is best described by viewing the onion routing process as a whole.

- The router at the message initiation point selects a number of onion routers at random.
- For each chosen router, the first router generates a message, provides

it with a symmetric key for message decryption, and specifies which router is next in the path.

- Each generated message is encrypted with the public key of the intended router.
- Thus a layered structure, or 'onion', is constructed. Each of the outer layers must be decrypted to reach the message at the inner-most layer

Figure 2.3 shows a graphic depicting an onion message on the left and to the right the path of this message through the onion routing proxies. Each square in figure 2.3 represents an onion routing proxy. The innermost layer of the onion message is for the final proxy destination of 2. From the original sender, proxy 1, three hops are required to reach the destination of proxy 2. Thus a further three encryption layers are added to the message. As the onion moves along the path the receiving router will decrypt a layer and send it to the next router in the path. This continues until the message reaches its final destination.

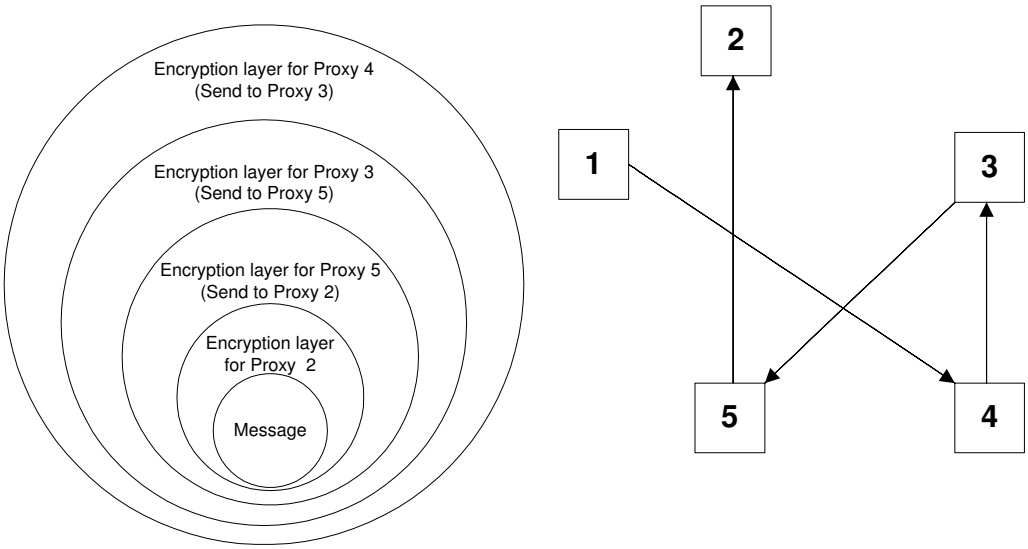


Figure 2.3: Onion Routed Message Layers and Message Hops

Thus anonymity tools can be very effective at protecting the identity of individuals as they transact on the Worldwide Web. However, there are times when a user may wish to maintain a persistent relationship with an Internet site, while still hiding their true identity. Anonymizing tools, by

their very nature render such relationships impossible. Pseudonymity, on the other hand enables the establishment of persistent, yet non-identifiable relationships.

2.4 Privacy Through Pseudonymity

A pseudonym is an identifier for a party to a transaction which cannot, under normal circumstances, result in the establishment of the true identity of an individual (Clarke, 1996). A Web transaction is said to be pseudonymous, when an individual can transact with no direct identifier to that individual present in the transaction data. The persistence of pseudonyms permits the establishment of long term web relationships without the loss of privacy (Rao & Rohatgi, 2000). Users may even choose to have multiple pseudonyms, used for a different set of activities. By so doing, their privacy can be further protected. A decided advantage of pseudonymity is that it holds benefits to users and site owners. Sites are able to offer personalized content to individuals based on their pseudonymous identity, and individuals can benefit from the personalized service without exposing their true identity. The Lucent Personal Web Assistant (LPWA) is a tool that implements pseudonymity and is examined briefly below.

2.4.1 The Lucent Personal Web Assistant (LPWA)

LPWA is a proxy that combines the features of anonymity and pseudonymity. It not only obscures the IP address of a browser initiating a request, but also allows users to enter alias information when registering at different Web sites. These aliases are managed in a manner to ensure consistent access at the registered sites (Abe, 1999). The basic operation of LPWA is depicted in figure 2.4. A proxy is located on a user's machine, with their Web browser set up to direct all Internet request to the proxy. As shown in figure 2.4, a request is received by the proxy, whereafter the IP Address of the current browser is removed. The request will then be forwarded to the requested site. Pseudonymity is provided when sites request registration information. When confronted with a registration page, users enter predefined codes in the userid, password and email fields. On submission of this data, LPWA will intercept those codes, and replace them with nonsensical pseudonyms.

These pseudonyms are used by LPWA to manage further interaction with the site.

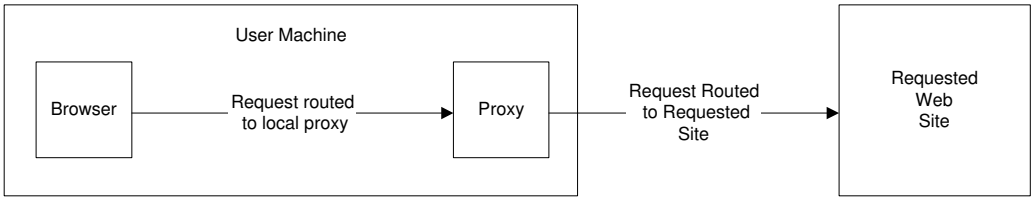


Figure 2.4: Pseudonymity with the Lucent Personal Web Assistant

Pseudonymity is not without its dangers. Should someone discover the real identity of a pseudonymous user, all of the users past actions are automatically exposed.

There is no doubt that anonymity and pseudonymity can play an important part in maintaining user privacy on the Internet. As an interesting aside, anonymity and pseudonymity also have roles to play in the arena of mobile communications. Mobile applications allow for the collection and storage of both the current location, and movement history of their users. This could lead to the profiling of users according to their movements. The purchase of a prepaid Subscriber Identity Module (SIM), where legally permissable, will enable a subscriber to communicate anonymously or pseudonymously (Rannenberg, 2004).

There are however, many instances when it may not be practical to transact anonymously or pseudonymously on the Web. In such cases, users do indeed have to surrender personal information. Privacy policies, discussed in the section below, can provide users with the control to at the very least, make informed decisions when releasing personal information.

2.5 Privacy Policy Specification

Anonymity and pseudonymity may not always be a solution to protecting privacy on the Web. Many transactions will require the release of personal information (Tavani, 1999; Olivier, 2003). Many Web sites have responded to the privacy concerns of users by publishing a human-readable privacy detailing their practices with regards personal information (Presler-Marshall, 2000). However, finding these policies can be time-consuming, and once

found the policy detail can range between two extremes. On the one hand policies may be so full of technical detail and legalistic complexity as to defy understanding, and on the other so lacking in detail as to prove useless (Presler-Marshall, 2000; Treese, 2000). This leads to the discussion of the Platform for Privacy Preferences (P3P) which aims at enabling Web sites to create privacy policies that can be located easily as well as interpreted.

2.5.1 The Platform for Privacy Preferences (P3P)

The Platform for Privacy Preferences (P3P) is an initiative of the World Wide Web Consortium (W3C). "It provides a standard way for Web sites to communicate about their privacy practices regarding the collection, use and distribution of personal information" (Cranor & Garfinkel, 2002). P3P provides users with greater control by not only enabling users to discover understandable privacy policies, but also to act on these policies (Presler-Marshall, 2000). A detailed specification of P3P 1.1 is available at (World Wide Web Consortium, 2002c). The operations of P3P will be discussed in greater depth in the subsections that follow.

How P3P Works

A P3P enabled site creates a machine-readable (XML) version of their human-readable privacy policy. This policy can be retrieved automatically by Web browsers and other user agents, and compared with a user's privacy preferences. If a site's P3P policy is in conflict with user preferences, the browser or agent can take steps necessary to avoid the offending data practice (Cranor, 2002; Softsteel Solutions, 2003). Thus P3P is dependent on a Web site component and a client component (World Wide Web Consortium, 2002a). A Web site converts their existing human-readable privacy policies, into a standard, machine-readable format(XML) for automatic retrieval and easy interpretation by user agents. On the client side is a P3P enabled user agent that automatically retrieves, interprets and provides user feedback regarding the P3P policies on Web sites. These agents either act proactively, for example blocking third party cookies, or more passively, for example informing a user of a discrepancy, but allowing them to decide whether or not to proceed. The P3P communication process between Web site and user agent is

illustrated by Figure 2.5 (Cranor & Garfinkel, 2002):

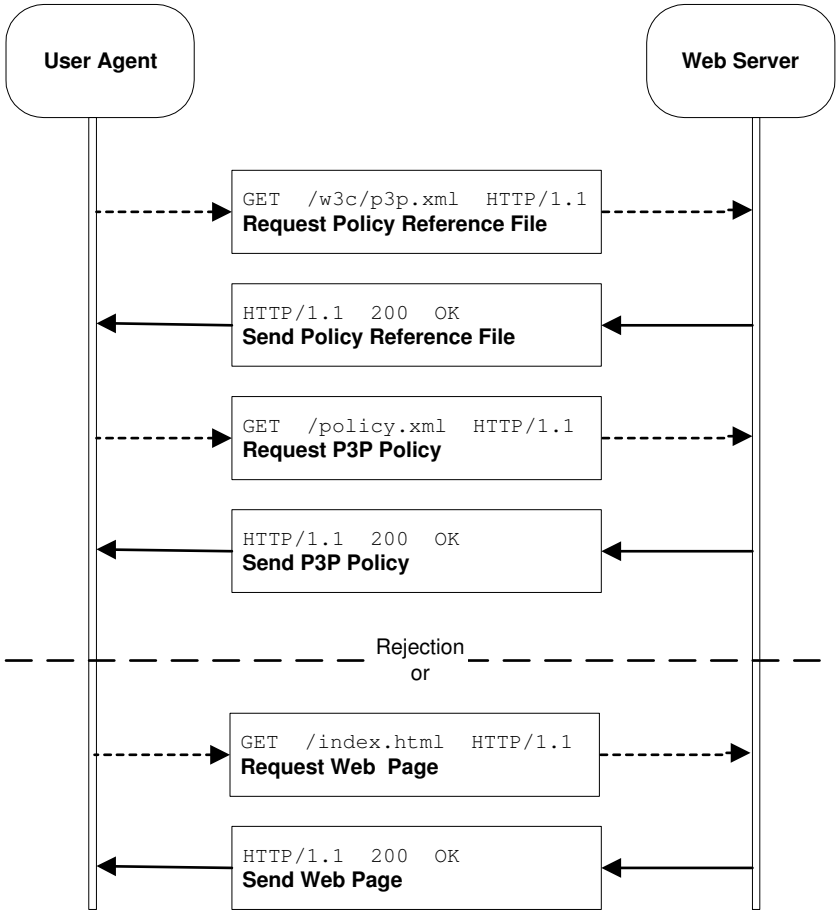


Figure 2.5: The Basic Protocol for Fetching a P3P Policy (Cranor, 2002)

- A P3P user agent requests a P3P policy reference file, using a standard HTTP request, from a well known location on the Web site of the requested resource.
- This reference file contains the location of the actual P3P policy file or files. There may be one policy for the entire site, or different ones applicable to different parts of a site.
- The user agent can then request the applicable policy file for interpretation, and take action according to the user's preferences or decision.

At the heart of P3P is the P3P Policy file. It is within this file that a Web site details its data practices regarding the data collected, how it is used with

whom it is shared and how long it is retained (Cranor, 2003; Presler-Marshall, 2000). Figure 2.6 shows a simple P3P policy file. P3P policies have eight major components. These components are expressed as XML elements and each may contain multiple subcomponents and attributes (Cranor, 2003). Table 2.1 provides a brief explanation of the eight major components of a P3P policy. A P3P policy should always contain a reference to the human-readable privacy policy. This is catered for by the *discuri* attribute of the `<POLICY>` element

```
<POLICIES xmlns="http://www.w3.org/2000/12/P3Pv1">
  <POLICY discuri=http://www.masters.com/privacy.html name="Masters Policy">
    <ENTITY>
      <DATA-GROUP>
       <DATA ref="#business.name">Masters</DATA>
       <DATA ref="#business.contact-info.online.email">masters@masters.com</DATA>
       <DATA ref="#business.contact-info.online.uri">http://www.masters.com</DATA>
      </DATA-GROUP>
    </ENTITY>
    <ACCESS>
      <all/>
    </ACCESS>
    <DISPUTES-GROUP>
      <DISPUTES resolution-type="service"
          service="http://www.masters.com/contact.html"
          short-description="Masters Contact">
        <REMEDIES>
          <correct />
        </REMEDIES>
      </DISPUTES>
      <DISPUTES resolution-type="independent"
          service="http://www.trustme.org/users/users_watchdog.html"
          short-description="TRUSTe">
        <REMEDIES>
       <correct />
        </REMEDIES>
      </DISPUTES>
    </DISPUTES-GROUP>
    <STATEMENT>
      <CONSEQUENCE>Masters keeps standard web server logs. We use this information
           for site administration and site improvements.  We will not
           disclose this information unless required by law. We retain log
           information indefinitely. Any questions can be directed to
           masters@masters.com
      </CONSEQUENCE>
      <PURPOSE> <admin /> <current /> <develop /> </PURPOSE>
      <RECIPIENT> <ours /> </RECIPIENT>
      <RETENTION> <indefinitely /> </RETENTION>
      <DATA-GROUP>
        <DATA ref="#dynamic.clickstream" />
        <DATA ref="#dynamic.http" />
      </DATA-GROUP>
    </STATEMENT>
  </POLICY>
</POLICIES>
```

Figure 2.6: A P3P Policy

The aforegoing focused on the creation of a P3P policy by a Web site. However of equal importance is the ability of users to express their privacy

Table 2.1: P3P Elements Explained

Element	Description
ENTITY	Contains information relating to the person or company declaring the privacy policy.
ACCESS	Specifies to what extent users have access to the information collected. Six types of access policies exist, ranging from all to none.
DISPUTES	Specifies the recourse open to users if they have a dispute over the site's privacy policy. The Service section provides information on how to settle disputes directly between users and the site itself. The Independent section provides details of third parties that users may contact to help settle dispute
CONSEQUENCE	General purpose description of the site's data practices
PURPOSE	Provides greater detail over exactly how collected information is used. Eleven purpose types exist with one *other-purpose*. Each purpose can have a required attribute set to indicate whether the purpose is *always* required or whether users can *opt-in* or *opt-out*.
RECIPIENT	States under what conditions data may be shared and whether users can *opt-in* or *opt-out*
RETENTION	States how long the collected information is stored. Five types of retention policies are available
DATA	Lists the type of data that is collected. Seventeen data elements, each with their own specific data elements exist.
STATEMENT	A Grouping comprising the purpose, data, recipients, retention and consequence elements. A Policy may have one or more statements. The data covered by each statement is treated the same.

preferences regarding privacy policies. To this end, P3P has a standard language for encoding user preferences - *A P3P Preference Exchange Language* (APPEL) (Cranor, 2003). It must be stressed that APPEL files were not designed to be read by end users. In most cases users would set up their preferences through a user agent, which itself may use APPEL. An example thereof is AT&T's Privacy Bird (Cranor, 2002), which allows users to specify their own privacy preferences which it creates using APPEL. It then compares these preference with a sites P3P-encoded privacy and alerts users when a site policy does not meet their standards. On accessing a site the

Privacy Bird provides a user with the following options:

- Provides a summary of a sites privacy policy.
- The ability to access a site's full human-readable privacy policy.
- The ability to access a site's page allowing users to opt-in or opt-out of collection practices.
- View the privacy policies of embedded content of the site, which may be different to the policy covering the page.

P3P also makes provision for optional compact policies, which are summarized P3P policies on how sites utilize cookies. Compact policies enable user agents to make snap decisions when applying user preferences with regards to cookies (World Wide Web Consortium, 2002c). At this point in time the two major Web browsers, namely Microsoft IE6 and Netscape Navigator 7 P3P support focuses primarily on compact policies.

Criticisms of P3P

P3P is not without its critics and numerous criticisms have been levelled against P3P, some of which are summarized below (Hochheiser, 2002; Softsteel Solutions, 2003).

- P3P provides no technical means by which privacy promises can be enforced.
- Many of the organizations that championed the cause of P3P did so in an attempt to avoid far reaching privacy protection legislation.
- P3P does not adequately address the principle of data collection limitation as laid down by such organizations as the Organization for Economic Cooperation and Development (OECD).
- The P3P vocabulary is open to ambiguities and confusion, for example the `RECIPIENT` element contains a value "ours", which is specified to mean "ourselves and/or entities acting as our agents or entities for whom we are acting as an agent" (World Wide Web Consortium, 2002c). It is conceivable that users might interpret "ours" to refer only to the primary Web site.

- P3P does not go far enough in protecting privacy and users might be forced into the use of a service provided by a Web site where no viable alternatives exist.
- The P3P vocabulary cannot describe privacy practices with the same level granularity as human-readable policies.
- Some supporters of P3P have oversold its benefits.

While many of these criticisms are indeed valid, some are perhaps unfair or at the very least distorted. For one, P3P was from the outset limited to, and aimed at, addressing notice and choice. As stated by Reagle and Cranor (1999), "We believe users' confidence in online transactions will increase when they are presented with meaningful information and choices about Web site privacy practices". The placement of machine-readable privacy policies for retrieval can be seen as notice, with the user's privacy preferences as a form of choice (Hochheiser, 2002). The W3C states clearly that P3P does not contend to solve all privacy concerns, nor does it negate the need for privacy legislation and privacy enhancing technologies. Rather P3P should be seen as complementary to legislation and other privacy tools (World Wide Web Consortium, 2002b).

Having now discussed P3P it is clear that whilst never intended, P3P does lack a mechanism to enforce privacy promises. The next section reviews work done in providing technological means of enforcing the privacy promises made in a P3P policy.

2.5.2 The Enterprise Privacy Authorization Language (EPAL)

The previous section specified how companies can publish their privacy promises as statements in a P3P policy. User agents can then automatically retrieve a policy and notify users whether they match their own privacy preferences. However, it was also stated that P3P does not possess technology to enforce privacy promises. The Enterprise Privacy Authorization Language (EPAL) is a formal language for the creation of enterprise privacy policies. An EPAL policy can enforce the privacy promises made in a P3P policy. A complimentary relationship exists between P3P and E-P3P. Whereas P3P allows

an organization to publish privacy policies in order to collect personally identifiable information from their customers; EPAL provides the much needed capability of privacy policy enforcement (Ashley, Hada, Karjoth, & Schunter, 2003; Karjoth, Schunter, & Waidner, 2002).

How EPAL Works

"An EPAL policy is essentially a list of privacy rules that are ordered with descending precedence (i.e., if a rule applies, subsequent rules are ignored)" (Ashley et al., 2003). An EPAL rule consists of a number of components and are described in table 2.2.

Table 2.2: EPAL Rule Components

Component	Description
Data User	Used to classify those individuals who either access or receive data.
Operations	Refers to activities that may take place against data e.g. read or create.
Categories	Defines the types of information an organization stores e.g. customer contact information
Purposes	Defines the purposes for which data may be accessed.
Conditions	The rules allowing access to data are governed by conditions e.g. the user must have consented before personally identifiable information can be used for a particular purpose.
Obligations	These are additional steps that may be imposed on data accesses e.g. log all data accesses for a particular purpose.

Informal privacy rules can be mapped to the more formal EPAL components. This is shown in Table 2.3. In addition to policy rules, two further sections are required in the formulation of an EPAL policy document (Schunter & Ashley, 2002):

Policy Information: This identifies the policy. It also contains version information, the date range between which the policy is valid and replacement policy information.

Definitions: This defines all of the components that can be used in the

EPAL rules. Data Users, Data Categories, Purposes, Actions, conditions and obligations, would all be defined here.

Rules: These are all the rules defining whether users are allowed or denied to perform actions on data categories, for which purposes and under what conditions.

Table 2.3: EPAL Example Rule Components (Ashley et al., 2003)

Component	Description
Privacy Rule	Allow a sales agent or a sales supervisor to collect a customer's data for order entry if the customer is older than 13 years of age and the customer has been notified of the privacy policy. Delete the data 3 years from now.
Ruling	Allow
Data User	Sales Department
Operation	Store
Category	Customer-record
Purpose	Order-processing
Condition	The customer is older then 13
Obligation	Delete the record 3 years from now

Comparisons to P3P

P3P is well suited to express the high level policies required by Web sites, but is not suitable for expressing internally enforceable policies. EPAL has been designed specifically for the specification of internal enforceable privacy policies (Schunter & Ashley, 2002). Another primary difference between P3P and EPAL, is that P3P has a predefined set of data categories, data users, purposes, operations, and choices. EPAL on the other hand, allows an organization to define its own list of each of these.

2.6 Conclusion

This chapter introduced and discussed the concept of privacy. People's concerns over privacy and the reasons for these concerns were also addressed.

Various technologies providing Internet users with a degree of anonymity or pseudonymity were discussed. However, it was made clear that anonymity and pseudonymity is not always possible or necessarily desirable, which lead to the introduction of policies as a means to give users greater control when having to release personal information. P3P was shown well suited to making an organization's intentions for personal information usage clear; but not capable of enforcing policy statements. This lead to the introduction of EPAL as a technological means whereby an organization can enforce its privacy promises. Just as an aside, it is the author's belief that despite the criticisms levelled against P3P it has at the very least stimulated the privacy debate. In fact the criticisms should serve as motivation for further research in the development of more encompassing privacy protection technologies. The Hippocratic log file proposal aims to serve such a purpose. Since this dissertation focuses on log files as the source of private information, the next chapter takes a more in-depth look at log files as a source of possibly private information.

Chapter 3

Log Files

The previous chapter addressed the topic of privacy and growing user privacy concerns, particularly when transacting on the Internet. Various technologies and mechanisms aimed at addressing these concerns were also discussed. These technologies and mechanisms include the use of anonymity and pseudonymity in protecting user privacy. However, it was previously emphasized that anonymity and pseudonymity is not always possible. In such cases users' personal information may be recorded.

This dissertation focuses on log files as such a collector of private and personal information. While it is true that many kinds of log files exist, the remainder of this chapter will primarily concern itself with those responsible for the logging of Internet activity. The chapter will take the following format. Section 3.1 will provide a brief set of definitions and overview of Web log files. Section 3.2 will identify the information that can and is recorded by Web servers. Some of the reasons for the recording of log file information is covered in section 3.3. Section 3.4 will argue why log files do indeed pose a privacy threat. The impact of privacy protection mechanisms, with respect to the recording of information in log files, will be examined in section 3.5.

3.1 Log Files

In the most general terms a log file can be defined as a file that maintains a list of actions that have transpired on a system. In many ways log files can be equated to an aeroplane's "blackbox", by the manner in which they provide a record keeping of system and network activity (Sarma & Mohirikar, 2003).

Log files pervade the computing environment. They can be found on stand-alone computers in operating systems, Web browsers, applications and E-mail; and they can be maintained by Web servers, databases and network devices to provide thorough tracking of all system activities. Each log file provides evidence of what has transpired within a system or application e.g. deleting an e-mail may not remove all trace of that e-mail, an e-mail log will contain the trail left behind by that e-mail.

The Internet has become a focal point for information logging. Every action of a user on the Internet is logged somewhere. One of the prime locations for logging user activity is the Web server log file. The Web server log file is a list detailing all the files it was requested to send and whether it was able to send them successfully (ExactTrend Software, 2001). The log files are stored as plain text and entries are generated every time a visitor accesses the Web site. When a user requests a page from a Web site, certain pieces of information are transferred from the server to the user's machine. The log file will contain one line of text for each "hit" to the Web site. A hit merely refers to the retrieval of a file from a Web site (Haynes, 2002). Since a Web page can include many graphics and other associated files, a request for a single page can result in multiple entries in the log file (ExactTrend Software, 2001). In other words each separate file accessed on a Web page, including HTML documents and graphics, count as a hit. The Web server log file will thus keep a very detailed account of the data transfer occurring between a user and the Web server. Various log file formats exist and logs may differ from business to business or indeed from Web site to Web site. However, regardless of the format or platform, most Web server log files record similar information. This will be viewed in greater depth in the next section.

3.2 Log File Information

Before one can argue whether log files pose a privacy threat it is important to understand the information recorded in the log file. The best way to explain this information is by means of a practical example. Below is a sample Web log file entry, followed by the name and description of each individual field within the entry. While the example used is specifically one of a Web server

log file, similar information is recorded by other log file types.

```
123.123.123.123 - - [25/Jun/2004:13:22:15 -400] "GET /cricket.htm
HTTP/1.1" 200 3456 "http://www.google.com/search/?q=cricket"
"Mozilla/4.0 (compatible; MSIE 5.5; Windows NT 5.0)"
"USERNAME=Andrew;TEAM=Proteas"
```

- **IP Address:** `123.123.123.123`
  This is the IP address of the machine that contacted the Web site and made a request.

- **Remote Username:** `-`
  RFC1413 defines a protocol used to determine the identity of a client that requests a resource from the server. This would be the user name of the client on the system from which they are connecting. It is seldom used on Internet servers, hence the first `'-'` following the IP address, indicating an unknown user name.

- **Authuser:** `-`
  When a user accesses password-protected content, this field will contain his username used for authentication. For normal unrestricted requests their is no username to record, hence the `'-'` indicated in this entry.

- **Timestamp:** `[25/Jun/2004:13:22:15 -400]`
  This is the time that the user issued their request as seen by the server.

- **Request:** `\GET /cricket.htm HTTP/1.1`
  The HTTP request line is recorded exactly as it comes from the user. In this case it was a 'GET' request for the file cricket.htm using the HTTP/1.1 protocol.

- **Result Status Code:** `200`
  This is the HTTP status code returned to the client. In this case the value `200` indicates success. If the requested file did not exist or could not be found then the code would have been `404`.

- **Bytes Transferred:** `3456`
  This is the number of bytes transferred from the Web server to the user. If it matches the actual file size it would indicate a successful

download. If the byte size was less it would suggest a failed or partial download.

- **Referrer URL:** `http://www.google.com/search/?q=cricket`
  The Referrer URL refers to the Internet location of the user immediately prior to requesting the current file. In this example the user came from a search engine page and as such the search criteria that the user used is also visible.

- **User Agent:** `Mozilla/4.0 (compatible; MSIE 5.5; Windows NT 5.0)`
  The user agent is a reference to the software that the user used to make the request as well as the operating system running on their system. Software making Web requests would usually be browsers but could also be Web robots, link checkers or FTP clients. The actual string that is recorded in the log file is set by the software manufacturer. In this example the user was using Microsoft Internet Explorer 5.5 running a Windows 2000 operating system.

- **Cookies:** `USERNAME=Andrew;TEAM=Proteas`
  Cookies are pieces of information that can be stored on a user machine by a Web server. When a user makes subsequent requests, the previously stored cookie information can be returned to the Web server. Thus, provided the Web server is configured to log cookies, cookie information will be logged.

It is thus clear that extensive information is stored within log files. Before considering the privacy implications of such information, it is necessary to identify reasons for its collection.

3.3 Reasons for Logging Information

Initially, the primary purpose of log files was to measure server load for diagnostic and planning purposes. However, increased network connectivity, the explosive growth of the Internet and consumer uptake of e-commerce have supplied additional reasons for the logging of information. The following sections review some of these uses for logged information.

3.3.1 Intrusion Detection and Computer Forensics

Due to the continued growth of Internet popularity, millions of people are now online. By creating a web presence a company may expose its internal network to these self-same millions. If only one percent of these users have malicious intentions, the security implications are substantial (Belgers, 1996). In many ways log files can be equated to an aeroplane's "blackbox" which records aircraft activity. In the event of a crash, this blackbox can provide valuable information that can help determine the cause of the crash. Log files provide a similar service by the manner in which they provide a record of system and network activity (Sarma & Mohirikar, 2003).

A connection to the Internet has forced most companies to implement firewall technology. Firewalls can also provide important logging and auditing functions, for example, logging the kinds and amounts of traffic passing through the network (Curtin & Ranum, 2000; Sarma & Mohirikar, 2003). Intrusion detection systems can make use of log file information as a data source, allowing them to identify tampering or malicious activity within a system. Once such activity is discovered, the log can provide valuable information such as the time of the attack, geographic location of the intruder and the break in-route of the intruder (Thomas, 2000). However, not only do log files allow the monitoring of possible intruders, but they can also be used to monitor the network activities of known users, such as employees.

3.3.2 Monitoring Employee Activity

The monitoring of employee activity is a contentious issue. The number of employees with Internet access has grown substantially in recent times. Along with this growth have come problems of decreased productivity, illicit communication of company secrets, and the accessibility to inappropriate material such as pornography (Nicolai Law Group P.C., 2001). Thus, privacy implications aside, employers have many valid reasons for using log file information to monitor employee activity. "Just as deadbolts and sophisticated alarms don't do much good if the thief is already inside the house, having computer network firewalls without monitoring employee activity can be equally ineffective inside the work-place" (Somerville, 2002). Actions taken by employees in misuse of the company network have definite implications

for their employers. These implications may be legal, for example, downloading pirate software, or economical, for example, using valuable company bandwidth for music streaming (Somerville, 2002). Although it is obviously important from an information security perspective to identify ill-willed behaviour from intruders and employees alike, log file information could also be very useful to learn more about visitors that a Web site may attract. This leads to a third use for log files: statistical analysis.

3.3.3 Statistical Analysis

It goes without saying that any company giving itself a web presence wants to attract visitors to its site – be this for reasons of e-commerce, or purely for the purposes of conveying their message to a larger audience. In order for a company to determine how effective its site is, it will need to track and measure their results. The web server logs capture valuable information that can be analyzed using a log file analyzer. The information that such an analysis yields includes most requested pages, least requested pages, top entry page, top exit page, single access pages, top referrer, search strings leading to a site, conversion rate of visitors to buying customers, and errors, such as links to non-existing pages. Only once it is known how visitors act on a site will it be possible to make the changes necessary to make the site more effective (Bailey, 2000; Internet Marketing Engine, 2001).

Statistical analysis need not require the use of personally identifiable information. Its major purpose is gaining an overall impression of the effectiveness of a Web site and pages within the site. However, personal identity could serve as a useful means of grouping related actions. Also, if companies want to adapt offerings based on individuals (or even groups), then that identity will have to be linked to future visits for the purpose of personalization.

3.3.4 Web Site Personalization

E-commerce continues to grow and likewise the competition amongst players in this arena increases. Personalizing the web experience for individuals holds great potential in winning new customers and increasing existing customer loyalty (Mulvenna, Anand, & Buchner, 2000). Personalization involves using information known about the user of a Web site, to customize that site

to better suit his needs or preferences. Thus, personalization requires the creation of user profiles, and Web log files provide an additional information source for the development of profiles. In addition, patterns in user navigational behaviour may be discovered, by applying data mining techniques to log file information (Eirinaki & Vazirgiannis, 2003).

3.4 Log File Privacy Threats

There is a question that needs to be answered clearly namely, "can users be personally identified by the information contained in log files?" Table 3.1, which represents some abbreviated log file data, will be used to help answer this question.

Table 3.1: Log File Entry

IP Address	Timestamp	Request	Referrer URL	Cookie
192.232.123.136	5/Jun/2004:13:22:15	/frogs.htm	http://www.google.com/search/?q=frogs	-
192.232.123.136	25/Jun/2004:13:23:25	/index.htm	http://www.google.com/search/?q=reptiles	-
192.232.123.136	25/Jun/2004:13:24:15	/toads.htm	http://www.reptiles.com/frogs.htm	user=fred@mail.com
192.232.123.136	25/Jun/2004:13:23:25	/frogs.htm	http://www.reptiles.com/index.htm	user=bob@mail.com
192.232.123.136	26/Jun/2004:13:26:25	/index.htm	http://www.anywhere.com	user=fred@mail.com

The most common piece of information that could be used to link back to an individual is the IP address. The data in table 3.1 is all for the same IP address. Making the assumption that this points to one individual would be incorrect in the majority of cases. Most people surf the Internet using an ISP, or from their place of work, quite possibly going through a proxy server. In such cases the IP address would be one assigned by the ISP or that of the proxy server itself. Thus, an IP address recorded in a log file could represent multiple users, as in the case of a proxy server. Two distinct IP addresses could well be a single user, assigned a different IP address for two distinct browsing sessions by their ISP. ISPs and proxy servers can of course keep a log of which user was assigned which IP address at any point in time (Kerkhofs, Vanhoof, & Pannemans, 2001). Were this information to be made available to a Web site contacted by a user, it would be a trivial task to correlate the provided information with their own log files and so personally identify the user.

Many Web sites provide certain services only if the users subscribe for their use. Users register by providing a username and password with which they may access the subscription services. Once users log into the site with their username and password, their username may be recorded in the Web log file for each subsequent activity, as shown in section 3.2. Thus, each log entry becomes personally identifiable, thereby facilitating the tracking of individual users as they navigate a Web site (Tec-Ed, Inc., 1999). However, when users subscribe to a site they have taken a conscious decision to do so. By providing true information they can be expected to be identified each time they return to the site. There are, however, means to identify returning users without affording them the opportunity of making such an informed decision.

Cookies can also be used as a means to identify individual Web users. On the first visit to a Web site, some form of personally identifiable information may be obtained from a user and stored in a cookie on a user's machine. In section 2.2, under the discussion of Internet specific threats, it was stated that cookies can only collect personal information voluntarily surrendered. However, it was also stated they may do so for one purpose, oblivious to the fact that it might be used for another. Suppose a user visits a site and is promised a monthly newsletter if they provide their e-mail address. The user may well be enticed to do so and in a snap that e-mail address can be placed in a cookie providing the information to personally identify that individual on a return visit. That cookie will now be returned to the creating server with each subsequent request. Observe the data in table 3.1, notice that as stated the IP address is the same for all entries. However, notice that once users have provided their e-mail addresses, the stored cookie is returned to the server for each request thereafter.

Another potential privacy threat posed by log files, is their use as a data source for the purposes of data mining. Cavoukian (1998) defines data mining as "a set of automated techniques used to extract buried or previously unknown pieces of information from large databases." Log files can contain detailed information of a user's online activities. Seen in isolation individual log file records might seem harmless enough. However, data mining algorithms allow such recorded data to be combined and analyzed, potentially creating user profiles (Tavani, 1999). When these profiles can be associated

with a particular individual, the implications may be significant. They may entail financial implications, for example, being placed in a high-risk insurance category with higher premiums, or just downright embarrassing, for example, men with a penchant for wearing baby nappies.

Thus, there are ways of using log files to personally identify individual users with definite privacy implications. How then can the tools and mechanisms discussed in sections 2.3, 2.4 and 2.5 protect users' privacy with regards to this information? Section 3.5 will briefly explore this question.

3.5 Log Files and Privacy Protection Mechanisms

Section 3.4 made it clear that there are ways of personally identifying individuals from log file information. With this in mind this section briefly revisits the mechanisms highlighted in sections 2.3, 2.4 and 2.5 to examine the role they can play in protecting individual privacy with regard to log file information.

3.5.1 Anonymity

The goal of anonymizing tools is to protect the privacy of users by making it difficult for Web sites to determine the source of a request. Anonymizing services typically submit requests to Web sites on behalf of its users. Because the request is submitted by the service, the only IP address recorded in a Web site's log file is that of the anonymizing tool itself. However, section 3.4 showed that in most cases it is difficult to identify an individual from an IP address alone, due largely to the manner most users surf the Web. Of course an IP address may not be able to identify an individual, but it can possibly identify the organization from which they gain access to the Internet. So removing IP addresses from requests can still play a part in privacy protection of log file information. It is worthwhile mentioning again however, that user requests are not anonymous to the anonymizing service or for that matter from their Internet service provider (Cranor, 1999; Treese, 2000).

A user using crowd technology rests assured knowing that the IP address recorded in a contacted Web site's log file, may be that of any member of

the crowd. However this may well be a double edged sword. Let's say for the sake of argument, a person in a crowd has an IP address that uniquely identifies them. This identity may be protected at the sites visited, because the chances are that the IP address recorded there will be that of another crowd member. But what about the requests made by other crowd members? Their requests can result in this persons IP address being recorded in a log file somewhere. What of the fact that a person may be identified at a site that he has never visited?

Onion routing is aimed at providing private communication between two parties, and on its own would not impact on the information that is recorded in a destination log file.

Of course users aiming for anonymity, will have to be very careful what information they release. If personally identifiable information is stored in a cookie, then protecting IP address information may be of little use.

3.5.2 Pseudonymity

Pseudonymity tools provide users with pseudonyms for use on sites requiring user names and passwords. Sites requiring registration may record user names in their log files. A user's privacy can therefore be protected if the recorded user name is in fact a pseudonym. Once again the information users release which may end up in a cookie, can negate all the benefits provided by pseudonymity.

3.5.3 Privacy Policy Specification

As mentioned previously many Web transactions require the release of personal information and in such cases anonymity and pseudonymity cannot be employed (Tavani, 1999; Olivier, 2003). Privacy policies will have no impact on what information is recorded in a log file as their aim is to inform users of information collection, as in the case of P3P, and to protect information after it has been recorded, EPAL.

The task of P3P policies is to inform users of what information is collected, and what the intended uses of that collected information may be. The manner in which P3P caters for defining the data collected in Web log files is of particular importance for this dissertation. P3P deals with log file infor-

mation using the dynamic data set, and specifically the clickstream and http data elements contained therein (P3PWriter, 2004). The clickstream element should apply to almost all Web sites and represents the information found in a standard Web server log file i.e. IP address, URI of requested resource, size of response etc. The http element caters for additional information within the HTTP protocol, for example, user agent information (World Wide Web Consortium, 2002c). Web sites can specify they collect all data contained within the `dynamic.clickstream` and `dynamic.http` elements, as shown in Figure 2.6. Alternatively, sites with more limited data practices can choose to list specific elements for example,

```
<DATA ref = "#dynamic.clickstream.uri" />
<DATA ref = "#dynamic.clickstream.clientip" />
```

An interesting fact regarding Web logs is that many small Web sites, hosted by third parties fail to make mention in their privacy policies that log data is collected. They themselves may not be collecting data, but the site host most likely will (P3PWriter, 2004). For the most part the P3P purposes of "Web Site and System Administration" and "Research and Development" will cover the collection purposes of log information. However, log file information can be combined with more personally identifiable information. In such cases additional purposes must be specified in the privacy policy, for example, the purposes of "Individual Analysis" and/or "Individual Decision" (P3PWriter, 2004).

3.6 Conclusion

This chapter explained the information most commonly recorded in log files, with a specific example of a Web server log file. Reasons were provided as to why this information recording takes place. Due to the fact that personally identifying information can be recorded, it is important to ensure that recorded information remains private. P3P policies can provide information to users regarding the information collected in log files. However, it was established that P3P does not cater for technological enforcement to ensure that collected information is used only for the intended purposes. EPAL was introduced in chapter 2 as a possible technological mechanism for policy enforcement. The chapters that follow address further such technologies.

Chapter 4

Hippocratic Databases

"And about whatever I may see or hear in treatment, or even without treatment, in the life of human beings – things that should not ever be blurted out outside –I will remain silent, holding such things to be unutterable" - Hippocratic Oath[1]

"For the dynamic, pervasive computer environments of the future, give end-users security they can understand and privacy they can control"
(Computing Research Association, 2003)

Chapter 2 of this dissertation highlighted privacy concerns, particularly when transacting on the Web. The Internet makes it a trivial task to automatically collect and store data; chapter 3 showed how log files can be used to harvest and process personal information. The privacy problem is further exacerbated by the fact that current tools for information collection and management, have not been designed to support the right to privacy. This technical oversight contributes greatly to the misuse of information collected from users of the World Wide Web (Bayardo & Srikant, 2003).

In a paper titled "Hippocratic Databases", Agrawal et al. (2002) outlined the concept of integrating the right to privacy within database management systems. Their proposed database system was inspired by the medical Hippocratic Oath, hence the term "Hippocratic Database". A founding tenet of a Hippocratic Database system is that it should be responsible for the privacy of data it manages. Ten principles of Hippocratic database systems have

[1]Translation by Heinrich Von Staden, In a pure and holy way: Personal and Professional Conduct in the Hippocratic Oath. Journal of the History of Medicine and Allied Sciences 51 (1996) 406-408.

been defined. The initial concept of Hippocratic database might well have been inspired by the Hippocratic oath, the outlined principles are, however, deeply rooted on the idea of "Fair Information Practices". These practices are themselves based on the privacy principles outlined by Organization for Economic Co-operation and Development (OECD) in 1980 (OECD, 1998). The Hippocratic designers further outlined a strawman design along with a set of use cases against which Hippocratic databases could be tested.

The remainder of this chapter serves as a summary discussion of the concept of Hippocratic Databases. This background knowledge is required before tackling the task of applying Hippocratic principles to log files. Section 4.1 will outline the 10 Hippocratic database principles. This will be followed by a section addressing the strawman design of a Hippocratic database architecture proposed by Agrawal et al. (2002). Section 4.3 will summarize some Hippocratic database challenges identified by the designers.

4.1 The Ten Hippocratic Principles

The OECD guidelines specify eight key privacy principles (OECD, 1998), while the designers of Hippocratic databases propose ten (Agrawal et al., 2002). Their ten principles aimed at governing the use, disclosure, retention and security of personal information. The ten Hippocratic principles are summarized below.

Purpose Specification *For personal information stored in the database, the purposes for which the information has been collected shall be associated with that information.*

The purposes for which information may be collected will vary from organization to organization. In large part purposes will be determined by the type of organization, the context and content of the collected information and the transaction that collected the information. For example, an online music seller may require a customer's purchase history to facilitate making recommendations, while a census bureau may require age and other demographic information for the purpose of population management and future economic planning.

Consent *The purposes associated with personal information shall have consent of the donor of the personal information.*

In other words, for every purpose for which information is collected, the donor should have the choice of either providing or denying consent. Providing consent may take the form of opting-in, i.e. no information can be collected until there is donor consent to collection, or opting-out, i.e. information is collected until consent to do so is denied by the donor. Thus, for online music sellers to use purchase history for recommendations, they would have to make provision for customers to either opt-in or opt-out of the recommendation purpose.

Limited Collection *The personal information collected shall be limited to the minimum necessary for accomplishing the specified purpose.*

The information that needs to be collected for any given purpose must therefore be carefully contemplated. Only information truly required to achieve a purpose may be collected. If customer purchase history is used to make recommendations, there can be no justification for collecting their telephone numbers for this purpose. Similarly a census collection for population management should not require details of the purchases made by an individual,

Limited Use *The database shall run only those queries that are consistent with the purposes for which the information has been collected.*

Thus, access to information must be carefully controlled to ensure that when a query is performed for a particular purpose, only the information collected for that purpose is returned. The marketing department of an online music vendor may issue a query for credit card numbers for the purpose of customer recommendations. Such a query should not be allowed to run since credit card numbers would not have been collected for the stated purpose.

Limited Disclosure *The personal information stored in the database shall not be communicated outside the database for purposes other than those for which there is consent from the donor of the information.*

In other words, users must provide consent not only for the purposes of collecting information, but also consent to with whom the information

may be shared. A customer agreeing to receive recommendations from the current music seller does not imply that his purchase history can be sold to other companies for marketing purposes. Such a practice would bring about another purpose for information collection which would again require user/customer consent. A patient on the other hand may well consent that his doctor share his medical history with medical colleagues, particularly if it might facilitate diagnosis or a cure.

Limited Retention *Personal information shall be retained only as long as necessary for the fulfillment of the purposes for which it has been collected.*

Once the purpose for which information was collected has been achieved, the information should be purged. A customers credit card number may be required to complete an online purchase. However once the purchase is complete, the credit card number is no longer required. Purchase history details, on the other hand, may be required for an extended period of time.

Accuracy *Personal information stored in the database shall be accurate and up-to-date.*

Thus, checks must be in place to ensure that all collected information, and any subsequent modification of information results in accurate and up-to-date data. For example, verifying a customers shipping address prior to saving the data.

Safety *Personal information shall be protected by security safeguards against theft and other misappropriations.*

This principle thus requires adequate security safeguards to prevent theft and other mischief. Authentication mechanisms will ensure that only legitimate users gain access to the database, whilst authorization checks will make certain that authenticated users only gain access to information to which they have been granted access. As an added protection, sensitive information might even be encrypted. All of these mechanisms combined, will play a vital role in ensuring the confidentiality and integrity of stored information, and hence, greater user privacy.

In other words individuals wanting access to data should be authenticated, sensitive data should be encrypted etc.

Openness *A donor shall be able to access all information about the donor stored in the database.*

The essence of this principle is to allow the donor to confirm what information is being stored in the database. Thus customers whose purchase histories are stored, should be given access to view for themselves, the information pertaining to their purchase history.

Compliance *A donor shall be able to verify compliance with the above principles. Similarly the database shall be able to address a challenge concerning compliance.*

Thus, a detailed accounting will have to be maintained of who accesses information, for what purpose and when the access took place. A customer of the online music store may wish to verify that information is only being used for the purposes to which consent has been given.

Having discussed the principles on which Hippocratic databases are based, the focus now shifts to an architecture which can be implemented to adhere to and enforce the 10 Hippocratic principles.

4.2 The Hippocratic Database Architecture

In a summary of current database systems, Agrawal et al. (2002), cite Ullman (1988) who considers two properties fundamental for a database system:

- the ability to manage persistent data; and
- the ability to access large amounts of data efficiently.

In addition to these two properties they further postulate that certain capabilities are universal to database management systems:

- Support for at least one data model.
- High level language support for data structure definition, data access and data manipulation.

- Concurrency control in the form of transaction management.
- Controls to ensure authorized data access and data validity.
- A means to recover from system failure with minimal loss of existing data.

In defining the concept of Hippocratic Databases, the designers were very clear on two points. Firstly, a Hippocratic database will need all of the capabilities available in current database systems. Secondly, in the interests of privacy preservation, efficiency, while still important, may not be the central focus. Instead, ensuring that data is used for the purpose for which it was collected will be the overriding concern. The strawman architecture (see Figure 4.1) outlined by the Hippocratic database designers, serves not as a blueprint, but rather as a road-map for future development on the path to the realization of a fully functional Hippocratic database system.

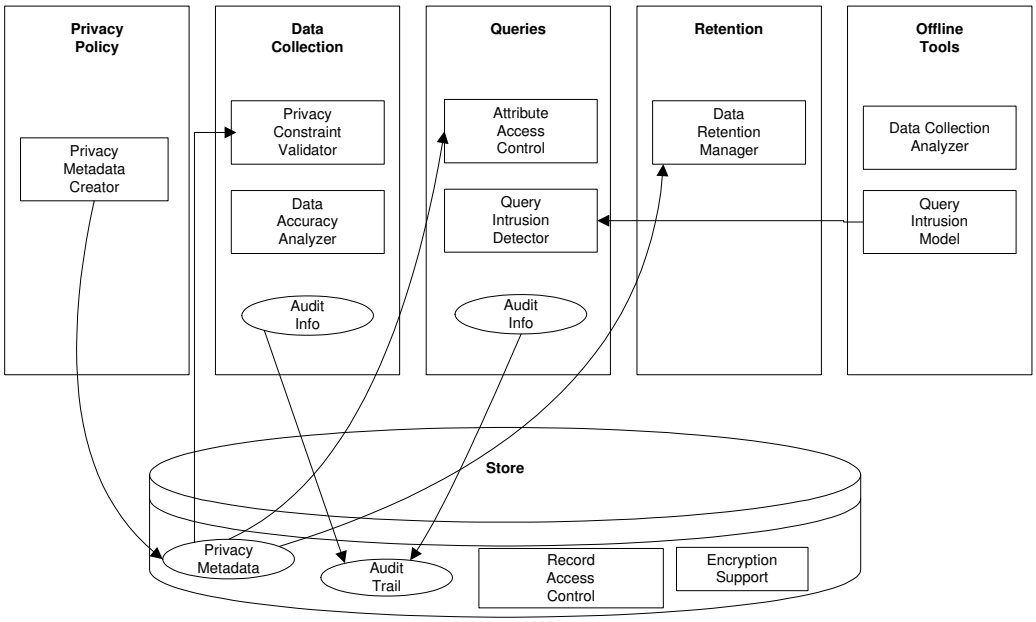


Figure 4.1: Strawman Architecture (Agrawal et al., 2002)

Each one of the major architectural components will be outlined in the subsections that follow.

4.2.1 Privacy Metadata

Privacy metadata tables are the means by which the purposes of data collection are defined. Each piece of collected information must be associated with the purpose(s) for which it is collected. Additionally, the following needs to be described and defined by the metadata:

- the *external-recipients:* with whom may this information be shared,
- the *retention-period:* the duration of time that the collected information is to be stored, and
- the *authorized-users:* the set of users and/or applications who may access the information.

Creating the metadata tables can be made easier by the privacy metadata creator. Its task would be to automatically generate the required metadata tables using the organizations privacy policy as its data source.

4.2.2 Data Collection

Prior to a user releasing information, the *Privacy Constraint Validator* will verify that the organization's privacy policy is in line with the user's privacy preferences. An *audit trail* of a user's acceptance of the privacy policy must be maintained to address any future challenges regarding compliance. Once user acceptance has been obtained, data can be inserted into the database. Along with each stored attribute, the purposes to which the user has agreed to must also be stored. In order to address the principle of accuracy, the *Data Accuracy Analyzer* should perform data accuracy checks. This may take place prior or after data insertion.

4.2.3 Queries

An *audit trail* of all queries must be maintained to address compliance challenges, as well as to enable external privacy audits. There are essentially three phases that take place in the fulfillment of a Hippocratic database query.

Before Query Execution: A check must take place to ensure that the person or application initiating the query is indeed among the list of authorized users. Once authentication has occurred, the *Attribute Access Control* unit should ensure that all of the requested attributes have a collection purpose matching the query's purpose. If a mismatch is detected the query should not be allowed to run.

During Query Execution: It is the task of the *Record Access Control* unit to ensure that only the data of users who have provided consent to the query purpose be returned. In other words, query results should be filtered of all users who have not provided consent to the query purpose.

After Query Execution: As with any technology that aims to address security or indeed privacy, there are always unscrupulous persons who may act unethically. An authorized user may issue a query for information and then attempt to use the information for their own personal gain or general mischief, for example, stealing customer email addresses to sell to direct marketers. To this end, a *Query Intrusion Detector* analyzes all query results to identify queries whose access pattern does not correlate with the normal access pattern of queries by that user for the given purpose. A *Query Intrusion Model* is used to model normal user access patterns and is used by the detector to identify potential misuse or intrusions.

4.2.4 Retention

The *Data Retention Manager* is responsible for deleting all information whose retention period has expired.

4.2.5 Additional Features

The *Data Collection Analyzer* will examine all queries for all purposes to determine:

- Any data collected but not used. In other words ensuring adherence to the principle of limited collection.
- Any data held for longer then required, thus supporting the principle of limited retention.

- Whether persons have unneeded authorizations for queries with a given purpose. This will play a vital role in ensuring the principles of limited use and limited disclosure.

4.3 Challenges to Hippocratic Databases

During the course of designing the strawman architecture the designers identified some problems and challenges. This subsection presents a summary of their findings.

4.3.1 A Policy and Preference Language

The specification of policies lies at the very heart of Hippocratic databases. The Hippocratic database designers believe that P3P and APPEL form a solid base for the expression of privacy policies and privacy preferences respectively. However, since P3P was geared towards the Web and Web shopping, they recommend building on the work of P3P to provide greater support for the richer environments in which they envisage Hippocratic databases operating. The efforts of Karjoth et al. (2002) are cited by the designers as work towards this end. Their work on EPAL was discussed in section 2.5.

4.3.2 Efficiency

Efficiency of record processing is an important aspect of modern database systems. This efficiency has been brought about by years of fine tuning record processing code. It has already been stated that processing efficiency will not be the primary focus of Hippocratic databases. However, efficiency cannot be completely ignored. A means must be found to process the extra overheads of protecting privacy in an efficient enough manner, so as to make Hippocratic databases viable.

4.3.3 Limited Collection

Ensuring that a database stores only the minimal information required to achieve all the purposes, and that queries access only the information required to fulfill its purpose, is a non-trivial endeavour. To achieve this goal three tasks need to be undertaken, each with their own challenges.

Access Analysis: Analyze the queries for each purpose to determine attributes that are collected for a purpose but not used. On the face of it, this seems like a simple matter of a union of all attributes mentioned in the set of queries for a given purpose. However, consider the following example. A mortgage application only needs to know about a user's assets when his salary is below a certain threshold. When the salary is not below the threshold it creates the impression that asset information is not required for the purpose. However, when the salary is below the threshold asset information is indeed required.

Granularity Analysis: Analyze the queries for each purpose, and for each numeric attribute, determine the granularity of information required. This can be motivated by the following example: a database may store the number of children that each of their customers have. If queries only ever ask "NumChildren > 0" or "NumChildren $= 0$", a case can be made that storing the value as a boolean provides sufficient granularity.

Minimal Query Generation: Generate the minimal query required to achieve the query goal. A certain amount of redundancy in application code may result from access analysis and granularity analysis – hence the need for minimal query generation.

4.3.4 Limited Disclosure

Allowing users the ability to dynamically choose the external recipients of their private information poses challenges for limiting disclosure. The Hippocratic designers show identity theft as one such problem. They propose that public-private key cryptography offers a possible solution, but concede deploying this solution poses its own challenges.

4.3.5 Limited Retention

Adhering to the principle of limited retention seems simple enough. On the face of it, it would appear that information should be deleted when it is no longer required. However, data is not only stored in the data table, but in the database logs and past checkpoints. Deleting data from these logs and checkpoints, without affecting recovery will be a challenge.

4.3.6 Safety of Information

Controlling the access to tables can primarily be controlled by the database system. However, the storage media on which the tables are stored may be vulnerable. For example, someone with super user authority may not have permission to access a table, but may gain access to database files using the operating system. While encryption of database files may help, the performance implications it entails will need serious consideration.

4.3.7 Openness

Even the principle of openness, which on the face of it appears easy, has its own challenges. Users should be able to determine if a database has information stored about them. However, in allowing this determination, the database should not know who issued the query, if they in fact hold no information of the querying user. Additionally, a user whose information is not stored, and who initiates a query for information, should learn nothing beyond the fact that no information is stored.

4.3.8 Compliance

Generating audit trails of every access to personal information and making this available to users, can be a powerful means to protect privacy. Doing this without paying a large performance penalty is a challenge. A potential solution may be the use of a trusted intermediary. Rather then sending the logs to each individual user, they may be sent to the intermediary. Users can then access log information on demand from the intermediary.

4.4 Conclusion

This chapter served to summarize the concept of Hippocratic Databases. Specific attention was given to the Hippocratic principles, an architecture for Hippocratic database implementation and challenges to its widespread adoption. The focus of the next two chapters is the answering of the fundamental question of this dissertation, namely: *"Can the privacy principles of Hippocratic databases be applied to log files?"*. Chapter 5 aims to address the applicability of the Hippocratic database principles to log files.

Chapter 5

Hippocratic Log Files: The Concepts

Hippocratic databases were discussed at length in the previous chapter. The focus of this chapter will be to investigate the applicability of Hippocratic database principles to log files. This will be achieved by discussing each of the Hippocratic database principles in turn, with a view to gauging the degree to which the principle could be applied to log files. Any challenges to full compliance with the principles will be highlighted, as will suggestions for adaptation in order to facilitate compliance.

5.1 Principles of Hippocratic Log Files

Discussing the conduciveness of the Hippocratic database principles is of primary importance to this dissertation. Demonstrating this applicability is required before any thought can be given to an architectural implementation of Hippocratic log files.

The format of the discussion will be as follows. Each principle as laid down by Agrawal et al. (2002) will appear in italics, with the word "database" substituted with the words "log file". After stating the principle, a short discussion of the applicability of that principle to log files will follow.

5.1.1 Purpose Specification

For personal information stored in the log file, the purposes for which the information has been collected shall be associated with that information.

There should be no problem in mapping this principle to log files. Chapter four of this dissertation presented a review of log files and included a section on valid reasons for information logging to occur. Associating these collection reasons with the personally identifiable information collected in log files, does not seem to pose a problem.

5.1.2 Consent

The purposes associated with personal information shall have consent of the donor of the personal information.

This is an admirable goal, and where possible the logging of personal information should adhere to it. However, there are reasons for logging information which supersede the right of the individual to consent to information collection – for example, intrusion detection and computer forensics. In the interests of openness, honesty and the Hippocratic spirit, the fact that this logging is taking place, should be communicated to users. Users can and should, however, be afforded the opportunity to provide consent for other purposes of information collection. For security reasons, a user should be logged while they are deciding whether or not to provide consent to these other purposes.

5.1.3 Limited Collection

The personal information collected shall be limited to the minimum necessary for accomplishing the specified purpose.

As stated previously, certain purposes for information collection override the users' right to consent. When collecting information for intrusion detection and computer forensic purposes the "minimum necessary" may indeed be as much as possible. One can argue that this still meets this principle's requirement, albeit that for this particular purpose the minimum information required is as much as possible. At this juncture it is important to stress that collection of information is a separate issue to the use of information, as addressed by the next principle.

5.1.4 Limited Use

The log file shall only permit queries that are consistent with the purposes for which the information has been collected.

A maximum amount of information may have been collected about a particular user for intrusion detection and computer forensic purposes. However, such information may not be used, say for purposes of statistical analysis, if the user has not given his consent to such usage. In this way the principles of limited collection and limited use, as they apply to log files, can operate harmoniously.

This particular principle will, however, place additional requirements on the manner in which most log files are currently stored. Log files should not be stored as un-encrypted plain text. Doing so would make it too easy for anyone with a text editor to view the information. Not all information contained within a log file is equally privacy sensitive. For example, a username has a much higher degree of sensitivity then say the date of access. Due to this fact, encryption might only be required for the more privacy sensitive log file information. A further storage requirement for log files is that they should be stored in locations that facilitate and enforce proper access controls. Access controls will need to ensure that only persons with the required access rights are granted access to log file information. Only in this manner can it be ensured that access to the log file takes place in accordance with the purpose of the information and the consent to use that information provided by the user.

5.1.5 Limited Disclosure

The personal information stored in the log file shall not be communicated outside the log file for purposes other than those for which there is consent from the donor of the information.

This principle overlaps with the principle of limited use, as disclosure of information goes hand in hand with the use of information. Once again, for this principle to be met, the issues raised in the previous subsection need to be addressed, i.e. encrypting log file information and restricting access to log information. During the course of a forensic investigation it may be required to disclose personal information, for example, the IP address of a potential

intruder. In terms of tracing offenders this seems reasonable. However, once the information is in the hands of a third party, they may use it for a purpose other than the one originally agreed to by the user.

Persons involved in computer forensics possibly need to undergo specialized training. This training might include the taking of an oath, prohibiting the disclosure of personal information other than for forensic purposes. Violation of this oath could result in the offender no longer being able to practice as a forensic professional. Another area that needs to be addressed is the scenario where a log file must be examined by law enforcement officials for the purpose of tracing a security offender. During the course of the investigation criminal activity unrelated to the initial security investigation may be discovered. In most legal systems this evidence would not be admissible in court. What this therefore implies is that the purpose for which access to log information is sought be clearly noted.

5.1.6 Limited Retention

Personal information shall be retained only as long as necessary for the purposes for which it has been collected.

When addressing intrusion detection and computer forensic issues, it may not be possible to specify an exact length of time for which the information is to be retained. However, the retention period should be reasonable. It has been stated that for security reasons, user information must be logged – even during the time when users are deciding whether or not to provide consent to other information collection purposes. In the spirit of Hippocratic databases, such information should be used for forensic purposes only, and should therefore be retained for a very short period.

For other purposes of information collection, where the user has consented to the use of information, the retention period can indeed be limited and information purged once the purpose of collection has been achieved. For purposes of statistical and trend analysis it may be required for information to be retained for extended periods of time. However, information can be aggregated and summarized, for example, by no longer storing information on each page hit, but merely the total number of page hits. This aggregated information can then still be stored and used for statistical and trend analysis, while no longer containing any personally identifiable information.

In the event that information is not summarized, it can be sanitized of personally identifiable information, once the reasons for needing this information have expired, for example, removing the IP address and usernames from all log records. Limited retention is very closely linked to the principle of purpose, i.e. the purpose for which information is collected will govern the retention period.

5.1.7 Accuracy

Personal information stored in the log file shall be accurate and up-to-date.

This principle, as it applies to log files, is a non-issue. Databases require the manual entry of information by humans. In such cases human entry errors will always be a concern. Log file information collection, on the other hand, is an automatic, machine-driven process and the same concerns of data accuracy do not apply, and hence, data accuracy is unimpaired. It must be remembered, however, that a machine will accurately record the information it receives, but has no way of verifying that this information is correct.

5.1.8 Safety

Personal information shall be protected by security safeguards against theft and other misappropriations.

The safety principle overlaps with at least two other principles, namely limited use and limited disclosure. In order for the safety principle to be met, previously raised issues need to be addressed, i.e. log files may need to be encrypted and access control mechanisms need to be in place to enforce users' privacy preferences.

5.1.9 Openness

A donor shall be able to access all information about the donor stored in the log file.

The issue that needs to be addressed here, is the degree of openness that is required as it applies to log files. Should the information that is collected for intrusion detection and computer forensics purposes be open for the donor to see? This would give the opportunity for intruders into the network, to view what information regarding their activities has been stored. This may

then give them the opportunity to attempt to cover their tracks before any intrusion is detected. Once again this highlights the need for the format in which log files are stored and accessed to be addressed. If all information regarding information donors is open for their inspection, then mechanisms must be in place to ensure that this information cannot be deleted or altered.

The information collected for purposes to which the user has consented, does not pose the same concerns as forensic information. Such information should be open for inspection. The ability of users to access log files raises a question – should such user access itself be logged? It has already been motivated that the information in the log file will be accurate due to the machine-driven nature of its collection. User inspection would thus primarily be to see what information is stored, and not to check the accuracy of information.

5.1.10 Compliance

A donor shall be able to verify compliance with the above principles. Similarly the log file shall be able to address a challenge concerning compliance.

Ensuring that the principle of compliance is met, raises new issues and challenges. A log file provides an audit trail of what has transpired on a system or network. This raises the question of whether one needs a log file to maintain an audit trail of accesses to that log file. What information would such a log file contain? If this audit log file contains personally identifiable information, will we once again need user consent for its collection? Another possibility is for log files to be stored by a party trusted by the information donor and the information collector. The role of this third party would be to ensure compliance to the principles of Hippocratic log files. The involvement of a third party in the process will itself raise new areas of concern, for example, if the connection to the third party is interrupted, then information that is required for security and forensic purposes will not be captured.

5.2 Conclusion

This chapter examined the application of Hippocratic database principles to log files. Hippocratic log files could serve as a means of providing users with

greater control over their personal information. Each of the ten Hippocratic principles were discussed. This discussion raised several issues relating to log files, that require resolution. These issues primarily revolve around the format in which log files are stored; plain, un-encrypted text is not a viable option. In addition, access control mechanisms to log file information must ensure and enforce user privacy preferences.

Table 5.1 summarizes the findings and opinions regarding the applicability of Hippocratic database principles to log files. Each row of table 5.1 contains a principle followed by a compliance indicator. A ++ indicates potential full compliance. A + indicates potential compliance, subject to limitations due to security and forensic purposes of log files. A - indicates that a principle is a non-issue. A ++* indicates potential full compliance, provided technical issues regarding the manner in which log files are stored and accessed are addressed.

Table 5.1: Hippocratic Log File Compliance

Principle	**Compliance**
Purpose Specification	++
Consent	+
Limited Collection	+
Limited Use	++*
Limited Disclosure	++*
Limited Retention	++
Accuracy	-
Safety	++*
Openness	+
Compliance	++*

The applicability of the Hippocratic principles to log files appears to be viable. There are indeed challenges to be overcome in order for full compliance with the Hippocratic principles. From the aforegoing principles discussion, the challenges facing Hippocratic log files parallel strongly those promulgated by Agrawal et al. (2002). This should not be surprising since data is data, whether it be stored in a database or a text file. From a security and access control point of view, log files (as they are currently stored) are indeed more vulnerable then databases. This is due to the fact of the richer security and access mechanisms built within modern database systems. Chapter

6 presents architectural issues related to the implementation of Hippocratic log files. Particular emphasis will be placed on how this architecture can enable users to maintain greater control over personally identifiable information collected unobtrusively.

Chapter 6

Hippocratic Log Files: An Architectural View

Chapter 5 discussed the applicability of Hippocratic Database principles, and concluded that this is indeed viable. This viability allows for the discussion and development of a Hippocratic log file architecture. Such an architecture should strive to provide users with security they can understand and privacy they can control. This would be in line with one of the four grand challenges for computing environments of the future, as identified by the Computing Research Association (2003). The envisioned architecture should shield users from technical details, while still placing them firmly in control of their private information.

From a user perspective then, it is important to know what personally identifiable information is stored in log files, the purposes for which this information is stored, with whom this information may be shared and the retention period for that information. The means provided for users to provide consent and specify privacy preferences should be user friendly and intuitive. By enabling user preferences to play a vital role in the collection and subsequent use of information gives them a greater degree of control of their privacy. The ease with which users can secure and control their private information, will aid in establishing trust relationships between users and information collectors. The technical architectural details, such as the physical storage locations of log files, or the mechanisms required to control access to log file information, can be well and truly hidden from users.

The two subsections that follow essentially depict two views of an ar-

chitecture for Hippocratic log files. The first is a high level overview of the interactions that would take place between users and Web sites in a Hippocratic-enabled World Wide Web. Secondly, a layered view of Hippocratic logs is presented which focuses on the primary processes that would be involved in such an architecture.

6.1 A High Level View of a Hippocratic Log File Architecture

Figure 6.1 represents a high level overview of a possible implementation of Hippocratic log files. This architecture was designed to conform to the principles espoused in section 5.1. A solid line on the diagram indicates an action that will occur, while a dashed line indicates an action that may occur, depending on user choices.

Users initiating requests would first be routed to an "unlogged" server which performs limited logging - this is indicated by the (A) in figure 6.1. The logs maintained by this server will be of a very temporary nature, for example, 24 hours. The idea of utilizing a completely unlogged server was considered, but rejected due to possible security implications. At this "unlogged" server, users will be informed that the logged server logs personal information for the purposes of security and forensics. It can be made clear to them that information collected for security reasons will only be used for security related purposes. Any other reasons for which collected information may be used, should be made clear. Thus, at this point users have the opportunity to terminate communication if they so desire. Due to the temporary nature of the logs on this server, any information they released will be discarded. This ensures compliance to the Hippocratic principle of consent.

As stated previously in this dissertation, users may be logged when initiating requests either from their place of work, or through an ISP. In such cases employers should inform employees of company logging policies and ISPs should do the same for their subscribers.

The (B) in figure 6.1 indicates the point at which user privacy preferences and consent choices come into play and is the main mechanism by which the consent principle is enforced. There are a few ways in which this interaction

could be managed.

- A user may have created a global set of privacy preferences to apply to all sites. In such a case, user preferences can be compared to the site's privacy policy and if there is no mismatch, could be routed directly to the logged server.

- A user may not have a global set of preferences or indeed a Web site may wish to create user preferences according to a model of their own choosing. In such a case, a site can inform users of the purposes of information collection, recipients of collected information, retention periods etc. Users can provide answers to presented questions, and in so doing their preferences can be created and saved. Cookies may play a role in this scenario. They would provide a means of recognizing return users, negating the need to re-enter preferences, and allowing return users to be automatically routed to the logged server. However, a means should be available for users to modify their preferences.

In figure 6.1(C) and (D) indicate a user being routed to the main server. By this time a user has agreed to the logging of information based on privacy preferences. All activity occurring on the logged server is recorded to the log file, as indicated by (E). All personal information that is logged will contain the purpose(s) for which it is logged, thus adhering to the Hippocratic purpose principle.

The (F) in figure 6.1 shows a request for log file information. Such a request could be from within the organization itself, or potentially from a user whose information has been collected. The degree of openness given to users, with regards to log file information, raises several architectural questions. In the first instance, should users be granted access to this information? Secondly, if access is granted, should access not be controlled by an additional server maintaining a copy of the log file? Thirdly, should all user accesses to the log be themselves logged? An alternative to allowing users full access to the log file would be to allow them access to the audit log only. In this manner potential intruders may not see what information is stored, but legitimate users will have a means to monitor when their information was accessed, and for what purpose.

Questions of openness aside, all requests, as indicated by (F) in figure 6.1, would pass through a log query processor. Part and parcel of the query processor's responsibilities would be to enforce access control mechanisms. These mechanisms would verify that the person requesting access is authorized to view the information. They will also ensure that the information returned or accessed, be restricted to those users who have consented to its use. By maintaining strict access control, adherence to the principles of limited use, limited disclosure and safety can be ensured.

All attempts to access the log file, successful or unsuccessful, should be logged to an audit log, as indicated by (G) in figure 6.1. Logging these accesses will aid in the enforcement of Hippocratic principles, particularly the principle of compliance. Information contained in the audit log will provide a history of who has accessed, or attempted to access, the log file. The purpose for which the log file was accessed will also be recorded. If access requirements are fulfilled, access to the log file(s) will be granted – indicated by (H). The audit log can be referred to if questions of compliance to Hippocratic principles are raised.

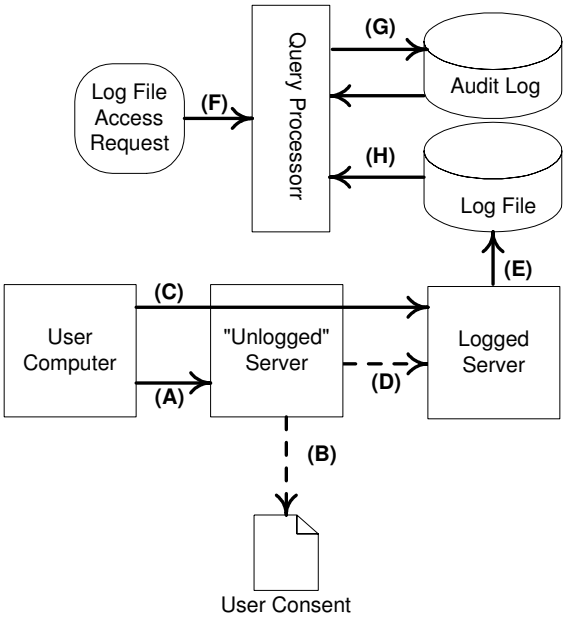


Figure 6.1: High-level Hippocratic Log File Architecture

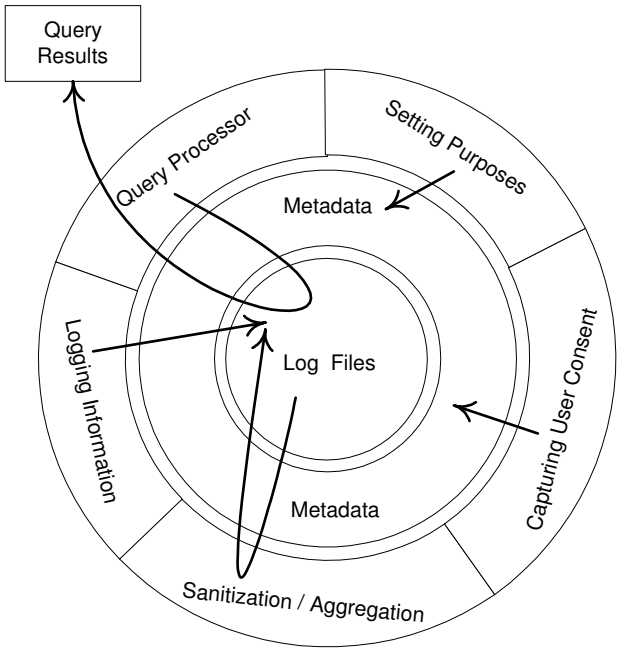


Figure 6.2: A Layered View of the Hippocratic Log File Architecture

6.2 A Layered View of a Hippocratic Log File Architecture

Figure 6.2 shows a layered view of the proposed Hippocratic log file architecture. The architecture has been abstracted to three layers. The first layer consists of the log files themselves. Log files are shown to be "surrounded" by the second layer, namely metadata. The function of this metadata would be to store the purposes of information storage, data recipients, retention period, as well as users' consent choices, with regards to the collected information. The details of how this metadata will be stored and formatted, be it in XML, database tables etc., falls beyond the scope of this dissertation. Log files are accessed either for information retrieval or information storage. Surrounding log files with a layer of metadata would be the primary means to ensure that all accesses to the log files, take place in accordance with user consent choices. In other words, consent metadata must be considered, before any access to log file information will be granted. The third layer, a functional layer, comprises the three major applications needing access to log files, as well as the mechanisms to capture consent and purpose metadata.

The following subsections will provide further procedural detail on each

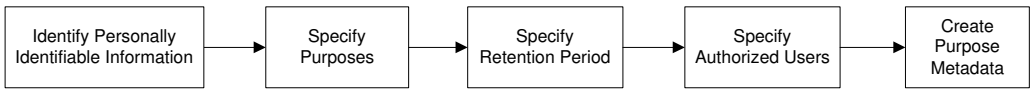


Figure 6.3: Setting Purpose Metadata

of the components of this functional layer. Many of the arguments and issues raised were inspired and influenced by the original Hippocratic database article of Agrawal et al. (2002).

6.2.1 Setting Up Purpose Metadata

Setting up purpose metadata would be the first step in establishing Hippocratic log files. This process is depicted by Figure 6.3.

Each piece of personally identifiable information that will be collected in log files must be identified. Once this has been done the purposes for which information will be collected as well as the retention period should be clearly defined and specified. A list of users/recipients who are allowed access to this information should also be identified. This list will enable strict access control to log file information.

Once all of the required purpose information is available, purpose metadata can be created and stored. As stated previously, the details of how this metadata will be stored and formatted, fall beyond the scope of this dissertation.

6.2.2 Capturing User Consent

The ability of users to provide consent to the use of log file information, for purposes other then those of security, is key to the concept of Hippocratic log files. As stated earlier in this chapter, users may themselves create a set of privacy preferences to be used for all Web interaction. In the event of sites wanting a greater degree over the creation of user consent metadata, the procedure would be as that depicted by figure 6.4. A user will make an initial request, for example, an attempt to access a Web site. If it is the first time visiting a site they will be routed to an "unlogged" server. The purpose of this server would be to provide a point at which users can view the reasons for information collection and the site's plans for its subsequent use. For

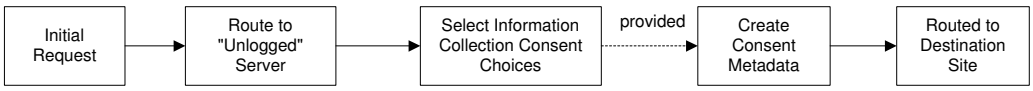


Figure 6.4: Creating User Metadata

security reasons, this server would itself need to log information. However, users should be informed that these logs will be kept for a very short period of time only. Thus, any user deciding to terminate communications at this juncture, can rest assured that their information will be discarded within a short while.

Users will further be informed of the need to record information for security reasons on the Web site which they requested. They will, however, be allowed to choose whether or not to consent to other collection purposes. Each of the collection purposes should be explained, and a mechanism provided whereby users can either grant or deny consent. Their consent choices need to be stored as consent metadata. This metadata will be used to ensure that any user information, subsequently stored in log files, will only be used for purposes to which users have consented. Once user consent metadata has been created, they can be routed to the site originally requested.

It is possible for employers to log the outgoing internet traffic of their employees, for example, by using a proxy server log. In the spirit of Hippocratic log files, such practices should be made known. The employment contract would be the ideal place for this disclosure, and also provide a means whereby employee consent metadata can be captured.

6.2.3 Logging Information

During this phase, users interact on a server and their information is captured and stored in log files. The Hippocratic principle of purpose specification, requires that the purposes for which personally identifiable information is collected, be stored along with the information. Exactly how purposes are stored with personally identifiable information, is an issue that needs to be resolved.

One alternative would be to store the purpose for which information is collected along with every physical occurrence of such information. Such an approach would of course greatly increase the physical size of log files, as well

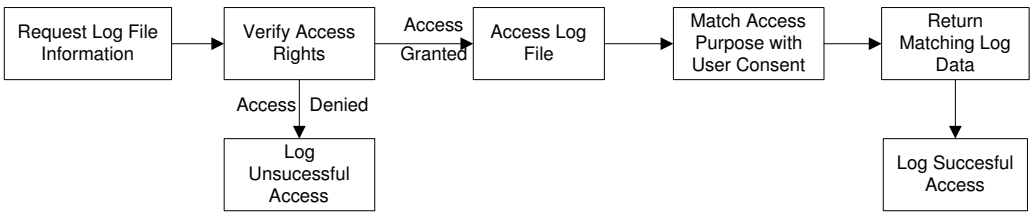


Figure 6.5: Log Query Analysis

as result in a great deal of purpose information repetition.

A further alternative might be to follow the approach as indicated in Figure 6.2. Purpose metadata can be created. In so doing each field of personally identifiable information in a log file, could be linked to this purpose metadata. This approach would minimize the storage implications that Hippocratic log files might impose, as well as avoiding unnecessary repetition.

Regardless of the approach implemented, the purpose for which information is collected, is a non-negotiable requirement and must be stored.

6.2.4 The Query Processor

The query processor will play a crucial role in ensuring that users' information is used only for the purposes to which they have consented. Figure 6.5 maps out the major functionality that a Hippocratic log file query processor would entail.

Any request to access log file information would be received by the query processor. The first task of the processor would be to verify that the person or process requesting information, has the required access rights to do so. Any request failing authentication would be denied access to log file information. The fact that there was an unsuccessful request will be logged for audit purposes.

Successful requests for security related purposes, would see the query processor draw the information directly from the log file. Security accesses will not be subject to any user consent metadata constraints i.e. user consent metadata need not be accessed when querying information for security related purposes. However, all other requests would require that the query processor interface with user consent metadata. During this interfacing, the query processor would ensure that only the information of users who pro-

vided consent for this particular purpose, be returned. Each successful log file access will itself be logged for audit purposes. A further task of the query processor will be to prevent the "privacy leak" problem identified by LeFevre et al. (2004). These leaks occur when information about individuals can be inferred from returned query results, despite adherence to user consent choices.

It was mentioned previously that storing log files as plain un-encrypted text poses problems. In the event of log files being encrypted, it would be a further task of the query processor, or an additional encryption/decription unit, to decrypt returned log file information.

6.2.5 Aggregation and Sanitization

Once the purpose for which personally identifiable information was collected has been achieved, it should be purged from the log file. This is in keeping with the Hippocratic principle of limited retention. The responsibility of ensuring limited retention, is housed with the sanitization/aggregation application of the architecture, as shown in Figure 6.2. This process would typically involve determining when the retention period for information storage has expired. Once this expiration is reached, there are three possible alternatives to remove personally identifiable information.

The first and most drastic option would be to delete the entire log entry i.e. delete personally as well as non-personally identifiable information.

A more moderate approach would be to aggregate log file information. During such a process all personally identifiable information would be removed and log file information would be summarized. Thus, for example, information on each page hit might be removed and replaced with the total number of page hits. The summarized information would still have value for high level statistical and trend analysis.

The final approach would be to keep the log entry but to put it through a process of sanitization. During this process log entries would be "de-identified". This would result in all identifying information, for example, IP address, usernames etc., being removed. All non-personally identifiable information would remain, and would still retain value for purposes such as statistical and trend analysis.

Previously it was mentioned that in the process of collecting user consent

metadata, users would be routed to an "unlogged" server. This server maintains log files of a temporary nature, for security reasons. If these logs are indeed to be temporary, then they would need to be aggregated and sanitized at a much faster rate then logs maintained by other servers.

6.3 Conclusion

Logging of personal information is a definite privacy concern for the owners of personal information. It has however been established that, particularly for reasons of security and computer forensics, information logging must take place. To alleviate user concerns, means need to be developed to minimize the privacy impact that this information holds. This can be achieved by giving users more control over their information. They may not be able to control its collection, but they can indeed have greater control over its use.

This chapter proposed two architectural views for Hippocratic log file implementation. These architectures serve a similar purpose to the original Hippocratic database strawman design in that they are not meant to be final designs, but rather to raise relevant issues and questions. Further investigation and research is required to further refine and develop these architectures. Chapter 7 will now present information of an exploratory prototype, developed to provide greater understanding and to practically demonstrate certain architectural components of Hippocratic log files. This prototype places particular emphasis on the role of the Hippocratic log file query processor.

Chapter 7

Exploratory Prototype

In chapter 6 of this dissertation several Hippocratic architectural issues were discussed. Two views of a Hippocratic log file architecture were proposed and are represented by Figure 6.1 and Figure 6.2. One of the fundamental components highlighted by these architectural views, was that of the query processor. The query processor was said to be responsible for the enforcement of log file access control. This enforcement would include ensuring that only authorized persons gain access to log file information, and then only in accordance with the consent choices of the information donors (users). This chapter presents findings of the development of an exploratory prototype, which focused on the simulation of one of the major components of a Hippocratic log file architecture; namely the query processor. The work of LeFevre et al. (2004) in a paper titled "Limiting Disclosure in Hippocratic Databases", proved invaluable in this regard.

7.1 Information Assumptions

The practice of storing log files as plain un-encrypted text has been identified as a hinderance to the achievement of Hippocratic log files. In lieu of this fact, it was decided to utilize a relational database as a storage mechanism for simulated log file information. This held a number of advantages:

- The purposes for which a company collects information can likewise be stored and linked to a log file table. In so doing users' privacy consent choices for information collection can be managed and maintained.

- Similarly, the recipients of log file information can be linked to purpose information, thus establishing who has access to log file information and for what purpose(s).
- Relational databases can be queried extensively using the Structured Query Language (SQL), i.e. a Hippocratic query processor can leverage this existing query language.

7.2 Information Representation

Having decided on a relational database as a storage medium, the actual table design structure needed to be defined. Figure 7.1 depicts an entity relationship diagram of the database design.

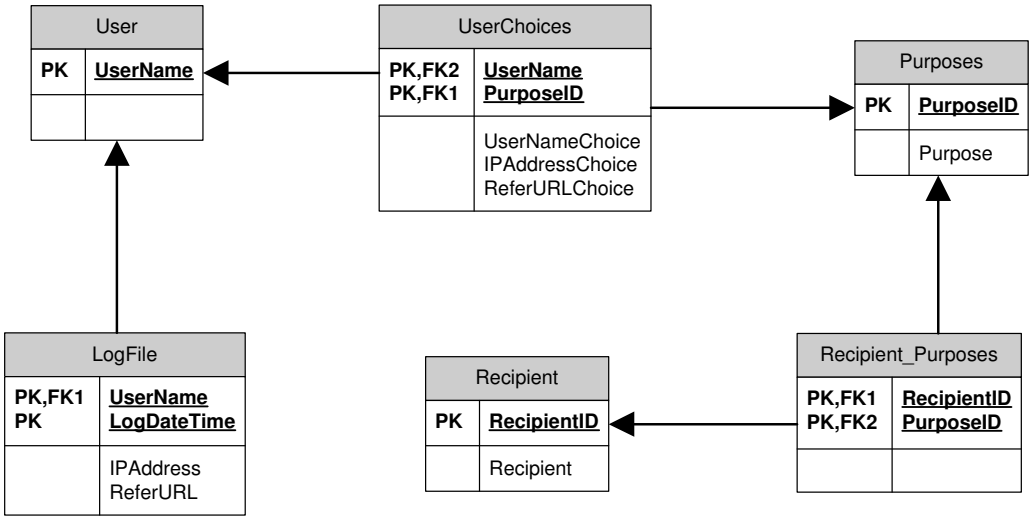


Figure 7.1: Entity Relationship Diagram

The User table depicted in figure 7.1 is a logical table used to establish a link between the LogFile and UserChoices tables and was not physically implemented.

The most important table would be that which would contain actual log file information. Table 7.1 represents this log file table with some sample fields and field descriptions. For the purposes of this experiment it was decided to utilize only a few of the fields currently catered for by today's log files. Adding additional fields is seen as an arbitrary exercise.

Table 7.1: LogFile Table Structure

Field Name	Field Description
UserName	The User name associated with current request
LogDateTime	The Date and time of current request
IPAddress	The Source IP Address of the current request
ReferURL	The URL of the user prior to the current request

The second table decided upon for this experiment, was one to store the purposes for which a company collects information. This is depicted in Table 7.2. This table will allow conformance with the Hippocratic principle of purpose specification.

Table 7.2: Purposes Table Structure

Field Name	Field Description
PurposeID	Identifier of a particular purpose
Purpose	Detailed description of the collection purpose

Having a log file table which will contain information regarding users requests (Table 7.1), and one storing the purposes for which a company collects log information (Table 7.2), it was necessary to have a table to store user consent choices regarding collection purposes. This is shown in Table 7.3.

Table 7.3: UserChoices Table Structure

Field Name	Field Description
UserName	The User name associated with current request
PurposeID	The PurposeID (associated with a purpose) for which information is collected
UserNameChoice	The user's choice regarding UserName field for this purpose
IPAddressChoice	The user's choice regarding IPAddress field for this purpose
ReferURLChoice	The user's choice regarding ReferURL field for this purpose

Table 7.3 will facilitate conformance to the Hippocratic principle of consent. Users can be given the opportunity to consent to the collection purposes contained in table 7.2 and their choices stored. These first three tables provide the primary means by which the use of a user's information can be

managed according to his consent choices.

However, of equal need for consideration is the Hippocratic principle of limited disclosure i.e. ensuring that log file information only be released to recipients who have authority to use this information, and then only for purposes for which they have authorization. This can be achieved by adding two additional tables to the relational design. The first is shown in Table 7.4 containing a collection of potential log file information recipients. A further

Table 7.4: Recipient Table Structure

Field Name	**Field Description**
RecipientID	Identifier of a Recipient
Recipient	Name of intended recipient or recipient group

table is required to control and maintain the purposes for which recipients may be granted access to information i.e. a link between the recipient table of Table 7.4 and the purpose table of Table 7.2. This link table is shown in Table 7.5 and serves to guarantee that information is only released to authorized recipients for specific purposes. Tables 7.6, 7.7, 7.8, 7.9 and 7.10

Table 7.5: Recipient_Purposes Table Structure

Field Name	**Field Description**
RecipientID	Identifier of a Recipient or recipient group
PurposeID	Identifier of a particular purpose

show all of the outlined tables loaded with sample data.

Table 7.6: LogFile Table with Sample Data

User Name	**Log DateTime**	**IP Address**	**Refer URL**
Andrew	03/Jun/2004:12:49	10.5.3.9	http://www.google.com/search?hl=en&q=Hamlet
Andrew	03/Jun/2004:19:49	10.5.3.9	http://www.google.com/search?hl=en&q=Shakespeare
Andrew	03/Jun/2004:19:50	10.5.3.9	http://www.google.com/search?hl=en&q=VB.NET
Reinhardt	03/Jun/2004:15:49	10.5.3.4	http://www.google.com/search?hl=en&q=Security
Reinhardt	03/Jun/2004:16:49	10.5.3.4	http://www.google.com/search?hl=en&q=Workflow
Werner	03/Jun/2004:13:49	10.5.3.2	http://www.google.com/search?hl=en&q=Latex
Werner	03/Jun/2004:13:55	10.5.3.2	http://www.google.com/search?hl=en&q=Alphabet

Table 7.7: Purposes Table with Sample Data

PurposeID	**Purpose**
SA	Statistical Analysis and Design
WP	Web Personalization
SF	Security and Forensics

Table 7.8: User Consent Table with Sample Data

UserName Choice	**PurposeID Choice**	**UserName Choice**	**IPAddress Choice**	**Refer URL**
Andrew	SA	Yes	Yes	No
Andrew	WP	Yes	No	No
Reinhardt	SA	Yes	Yes	No
Reinhardt	WP	No	No	Yes
Werner	SA	No	No	No
Werner	WP	Yes	No	Yes

Table 7.9: Recipient Table with Sample Data

RecipientID	**Recipient**
1	Sales
2	Marketing
3	Security Officer
4	Third-Party

Table 7.10: Recipient_Purposes Table with Sample Data

RecipientID	**PurposeID**
1	SA
2	WP
2	SA
3	SA
3	SF

7.3 Query Processor Implementation

Having catered for the information representation requirements of a Hippocratic log file, the next step was to implement the query processor itself. The query processor would be the technological enforcement mechanism of Hippocratic principles. For the purposes of this prototype the query processor was simulated by the creation of a small Visual Basic.NET application. The task of this application was to preprocess all queries of Log file information to ensure that Hippocratic log file principles are strictly enforced. The major processing steps are listed as follows:

1. An initial request for log file information is received from a recipient for a given purpose.

2. An access control check will ensure that the recipient indeed has rights to access information for the specified purpose.

3. Full access to log file information will be provided if the recipient has rights to access information for security purposes.

4. Accesses to log file information, for purposes other then that of security, where the recipient does indeed have access rights for the particular purpose, will be granted access. However, their request query will first go through a process of query modification. Case statement modification will be applied to the query to ensure that only the information of users who have granted consent to the current purpose is retrieved.

Concrete examples will now be used to illustrate the process specified in the preceding list.

Suppose the following query is issued by a recipient in the Sales department for the purpose of security and forensics.

```
SELECT IP_Address, UserName, LogDateTime, ReferURL FROM LogFile
```

By querying the Recipient_Purposes table it can quickly be established that someone in the Sales department does not have access to information based on a purpose of security. Such a check might be:

```
SELECT COUNT(PurposeID) FROM Recipient_Purposes
    WHERE RecipientID = 1 AND PurposeID = 'SF'
```

If the count returned by this query is zero, then the recipient has no rights to access the information. The same request for information issued by a recipient in the Security department would return a count of more than zero in the query, which would grant the recipient access to the information. The returned results are shown in Figure 7.2

Results of Query Processing

UserName	IP_Address	DateTimeLogged	ReferURL
Andrew	10.5.3.9	03/Jun/2004:12	http://www.google.com/search?hl=en&q=Hamlet
Andrew	10.5.3.9	03/Jun/2004:19	http://www.google.com/search?hl=en&q=Shakespeare
Andrew	10.5.3.9	03/Jun/2004:19	http://www.google.com/search?hl=en&q=VB.Net
Reinhardt	10.5.3.4	03/Jun/2004:15	http://www.google.com/search?hl=en&q=Security
Reinhardt	10.5.3.4	03/Jun/2004:16	http://www.google.com/search?hl=en&q=WorkFlow
Werner	10.5.3.2	03/Jun/2004:13	http://www.google.com/search?hl=en&q=Latex
Werner	10.5.2.2	03/Jun/2004:13	http://www.google.com/search?hl=en&q=Alphabet

Figure 7.2: Query Results for Security Purpose

In this case all information from the log file would be returned, provided the recipient has access to information for security reasons. There is no need to further pre-process the query since users do not have the right to consent to the use of their information for security purposes.

Now let's suppose a query is issued by a recipient in the Marketing department for the purpose of personalization. Assuming they are only interested in the user name and the referring URL used by users, the initial query may look thus:

```
Select UserName, ReferURL from LogFile
```

Once again a check will be done to verify that the intended recipient can access information for the specified purpose. Thereafter the query must be further processed to ensure that the information retrieved includes only the details of users who have provided consent for this purpose. The original query would be transformed into the following:

```
 SELECT
CASE WHEN EXISTS
  (SELECT UserNameChoice FROM UserChoices
    WHERE LogFile.UserName = UserChoices.UserName AND
    UserChoices.UserNameChoice = 1 AND PurposeID = 'WP')
    THEN UserName ELSE null END as UserName,
 CASE WHEN EXISTS
  (SELECT ReferURLChoice FROM UserChoices
    WHERE LogFile.UserName = UserChoices.UserName AND
```

```
    UserChoices.ReferURLChoice = 1 AND PurposeID = 'WP')
    THEN ReferURL ELSE null END as ReferURL
  FROM LogFile WHERE EXISTS
     (SELECT UserNameChoice FROM UserChoices
       WHERE Logfile.UserName = UserChoices.UserName AND
       UserChoices.UserNameChoice = 1 AND PurposeID = 'WP')
```

Each requested field from the original query is "wrapped" within a *CASE* statement. This statement verifies the accessibility to each field based upon the users' consent choices. If a user has given consent for its use the field value is returned. A Null value is returned where consent is lacking. The results of the pre-processed query are shown in Figure 7.3

Results of Query Processing

	UserName	ReferURL
▶	Andrew	(null)
	Andrew	(null)
	Andrew	(null)
	(null)	http://www.google.com/search?hl=en&q=Security
	(null)	http://www.google.com/search?hl=en&q=WorkFlow
	Werner	http://www.google.com/search?hl=en&q=Latex
	Werner	http://www.google.com/search?hl=en&q=Alphabet
*		

Figure 7.3: Query Results for Personalization Purpose

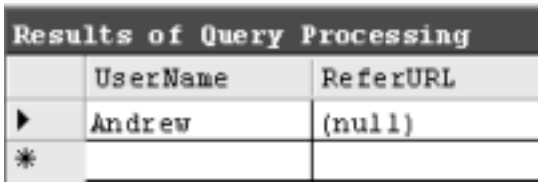
Referring back to Table 7.8 it can be seen that user "Andrew" did not give consent to the use of his referring URL field for personalization purposes, but did so for his user name. User "Reinhardt" provided the necessary consent for the referring URL field but not for his user name. User "Werner" provided consent for both requested fields.

The preceding examples and Table 7.3, clearly demonstrate how user consent choices can be defined and enforced for each individual data field of the log file. However, the problem of privacy leaks, raised by LeFevre et al. (2004), can arise, if the Query Processor is implemented as in the previous example.

Suppose someone in marketing runs a query, for the purposes of personalization, to see which users have been searching for a specific word, for example, 'Shakespeare'.

```
Select UserName,ReferURL from LogFile
   WHERE ReferURL LIKE '%Shakespeare%'
```

The results of this query, after modification by the Query Processor, are shown in Figure 7.4. The returned results do indeed conform to user "Andrew's" consent



Results of Query Processing		
	UserName	ReferURL
▶	Andrew	(null)
*		

Figure 7.4: Query Results for Personalization Purpose

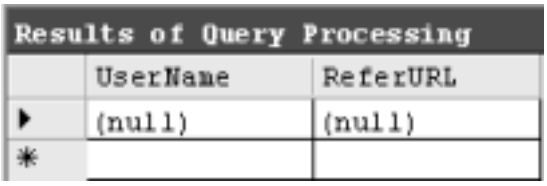
choices. He gave consent to the use of his user name for personalization purposes but not his referring URL field. Hence his user name is displayed in the query result set while his referring URL is omitted. However, despite the fact that the returned referring URL has been nullified, the nature of the query makes it possible for anyone to infer that "Andrew" has indeed previously searched for 'Shakespeare'. The reason for this 'leakage' of information in this scenario, is that a personally identifiable field was returned along with the nullified field.

In this experiment, the "privacy leak" was corrected by examining the WHERE clause of any query of log file information. Thereafter, when processing the query to return results, a user name (or any other personally identifiable information) will only be returned, if the user has granted consent to the use of all fields listed in the WHERE clause. Below is the original personalization query, which has been modified by the Query Processor to eliminate the "leakage" of information. The output of this query is shown in Figure 7.5.

```
SELECT CASE WHEN EXISTS
  (SELECT UserNameChoice FROM UserChoices
    WHERE LOGFILE.UserName = UserChoices.UserName AND
    userChoices.UserNameChoice = 1 AND PurposeID = 'WP' AND
    ReferURLChoice = 1)
    THEN UserName ELSE null END as UserName,
 CASE WHEN EXISTS
  (SELECT ReferURLChoice FROM UserChoices
    WHERE LogFile.UserName = UserChoices.UserName AND
    userChoices.ReferURLChoice = 1 AND PurposeID = 'WP')
    THEN ReferURL ELSE null END as ReferURL
 FROM LogFile WHERE EXISTS
   (SELECT UserNameChoice FROM UserChoices
     WHERE Logfile.UserName = UserChoices.UserName AND
```

```
UserChoices.UserNameChoice = 1 AND PurposeID = 'WP') AND
ReferURL like '%Shakespeare%'
```

Notice that within the CASE modification statement of the user name field, a further condition has been added. In this example the user name will now only be returned, if the user has indeed given consent to the use of his referring URL field for personalization purposes.



Results of Query Processing

	UserName	ReferURL
▶	(null)	(null)
*		

Figure 7.5: Query Results – Privacy Leak Elimination

7.4 Conclusion

During the course of this practical experiment it was possible to simulate the operations of a proposed Hippocratic Log File Query Processor. A relational database was used as a storage mechanism for Hippocratic Log file information. This included information pertaining to the purposes of information storage, user consent choices regarding purposes and the recipients of log file information. All queries of log file information were made using standard SQL. The Query Processor simulator could then verify that the intended recipient possessed the required rights to view the requested information. By a process of CASE statement modification, queries could be transformed to ensure that only the information having the consent of users for the specified purpose would be returned.

The next chapter will reflect on what this dissertation has achieved.

Chapter 8

Conclusion

Concerns for individual privacy are not new, but the rapid growth of the Internet has certainly intensified these concerns. Under certain circumstances, privacy on the Web can be maintained with the use of tools that provide anonymity or pseudonymity features. However, the nature of many Web transactions make the use of such tools infeasible. Thus, the inability to use such tools can result in an individual's personal information being collected and recorded, often without his knowledge or consent. This dissertation focused on log files as a collection and storage point of personal information.

No one can deny the need for information logging for security and computer forensic reasons. However, it is equally undeniable that the logging of personal information raises privacy concerns for the owners of that information. As stated earlier, the Computing Research Association (2003) has identified, as one of their four grand challenges, that the computing environments of the future must strive to *"give end-users security they can understand and privacy they can control"*. This dissertation aimed to make a contribution to this challenge, by attempting to demonstrate how users can be given greater control over the information collected unobtrusively by log files. In order to make this contribution, this research required background knowledge in certain key areas, as well as the answers to questions voiced in chapter 1. These will now be reviewed in order to measure the contribution that this dissertation has made to user privacy control

8.1 Research Questions Reviewed

Before delving into the fundamental research question of this dissertation, it was necessary to have a solid understanding of the domain of discourse. The three primary subjects of this domain were privacy, log files and Hippocratic databases.

Chapter 2 served to provide the background knowledge in the area of privacy. It began with a discussion on some of the definitional aspects of privacy. From this discussion it was concluded that despite differing opinions in terms of definitions, researchers in the field concur on importance of privacy in everyday life. Chapter 2 further reviewed reasons why the Internet has heightened user privacy concerns, and discussed mechanisms currently available for preserving user privacy. Of particular importance to this dissertation was the point made that at times it is impossible to avoid the recording of personal information. One of the collection and storage points of personal information is log files, which were discussed in chapter 3.

The discussion of log files in chapter 3, began by highlighting the type of information collected by log files, and discussed several reasons why this collection takes place. The privacy threats that such collected information pose, particularly when the use of privacy protection mechanisms is not possible, was highlighted.

Chapter 4 introduced the concept of Hippocratic databases, which was inspired by the medical Hippocratic oath, and promulgated by Agrawal et al. (2002). Discussion in this chapter showed that a Hippocratic database is to be responsible for the privacy of the data it manages. The 10 key Hippocratic database principles were thoroughly reviewed, as they served a fundamental purpose for this dissertation. Discussion in chapter 4 included a strawman architecture for Hippocratic database implementation, as well as certain challenges to Hippocratic databases expressed by the designers.

Having completed the discussion of all the required background knowledge, it was possible to begin answering the questions posed by chapter 1.

8.1.1 Can the privacy principles of Hippocratic databases be applied to log files?

This question was answered in chapter 5. This was achieved by reviewing each one of the Hippocratic principles in turn. The discussion undertaken, served to determine the applicability of each principle to log files. Challenges to compliance were discussed and suggestions for adaptation to achieve compliance, made. The primary challenge to overcome was identified as the current manner in which log files are currently stored i.e. un-encrypted plain text. Such a practice has negative implications on many of the principles, not least of which the principles of *limited use* and *limited disclosure*. However, none of the identified challenges seemed insurmountable and the conclusion drawn was that the Hippocratic principles are

indeed conducive to log files. This was summarized in table 5.1 on page 61.

8.1.2 How can the goal of giving users greater control over their private information be realized?

Having established the applicability of Hippocratic log files, the dissertation could move forward to argue how such log files could be implemented. Chapter 6 set out to define a Hippocratic log file architecture. This architecture was presented in two views. The first view highlighted a high-level functional architecture. The fundamentals of this architecture were discussed in a manner that demonstrated the control users could gain over the information collected by log files. The second view presented the architecture as a series of layers. The central layer showed log files surrounded by metadata. The major processes, governed by this metadata were discussed, highlighting the role that each would play in a Hippocratic log file implementation. Chapter 7 discussed the development of an exploratory prototype used to gain more understanding of Hippocratic log file concepts, and to practically demonstrate certain of the architectural components. By simulating a query processor, the prototype showed how users could maintain a degree of control over the use of their information stored in log files.

8.1.3 Given that anonymity on the Web is not always possible, by what other means can user privacy be assured?

Chapter 2 discussed the use of policies and their role in privacy protection. P3P was shown to be a good mechanism for informing users of Web sites' information collection practices. While P3P allows users to make informed choices about whether or not to provide their information, it lacks mechanisms to technologically enforce privacy promises. EPAL was reviewed as a policy language enabling the enforcement of promises made in P3P policies.

Considering the statements in section 8.1.2, this dissertation also demonstrates the effective role Hippocratic log files could play in situations where anonymity is not possible. Hippocratic log files would allow users the opportunity to provide or deny consent to the use of their information for stated collection purposes. Thus, in instances where anonymity is not possible, Hippocratic log files can yield to users, a substantial degree of control over their collected information.

8.1.4 What impact will the application of Hippocratic principles have on the unobtrusive collection of information by log files?

Hippocratic log files will initially cause a disturbance in the unobtrusive collection of information by log files. This is due to the fact that such log files will require that those doing the logging inform users of this fact, and specify the purposes for which information is logged. Additionally, users will have to be given the opportunity of providing or denying consent to collection purposes.

As stressed earlier, consent cannot be denied when logging information for security purposes. However, in the Hippocratic spirit, the fact that collection takes place for this purpose must still be conveyed to all users. Section 6.1 on page 64 discussed the concept of an "unlogged" server. The logs maintained by this server will be of a very temporary nature, and used purely for security reasons. At this "unlogged" server, users are informed of information collection and afforded the opportunity to provide or deny consent to the various collection purposes.

Although informing users initially of information collection interrupts the normal unobtrusive collection of information, once user consent choices are recorded, information logging can continue unobtrusively. However, users now have detailed knowledge of what information is collected and can rest assured that the use of this information will be controlled according to their consent choices.

8.2 Challenges and Future Work

Making a contribution to greater user control over their private information was fundamental to this research. Hippocratic log files were presented as a means to providing such control. There are however a number of issues requiring further research.

The practice of storing log files as plain, un-encrypted text will have to be replaced by an alternative means. The WinFS file system proposed by Microsoft, and which is implemented using database technologies, might address some of these storage issues. However, the envisioned ability of WinFS to allow users to search and manage files based on content, may pose interesting privacy challenges of its own. In WinFS the further trend to semantically link apparently disparate information as portrayed by, for example, the proponents of the Semantic Web, is also evident in-the-small. In fact the Semantic Web's effect on privacy needs to be investigated.

The Semantic Web of the future aims to move from a Web of linked documents to one of linked information. Its vision of information with well-defined meaning, will make it possible to link information in richer ways. The ability to link to log file information in these new ways, can only increase the privacy threat they pose.

The architecture presented in this dissertation is by no means complete. It served an important purpose in demonstrating a high-level view for Hippocratic log file implementation. However, further investigation and research is required to further refine and develop this architecture.

8.3 Final Words

Sun Microsystems CEO Scott MacNealy stated in 1999, "You have zero privacy, get over it". Such a defeatist view is appalling to this author. Privacy protection may have become increasingly difficult, but a wholesale capitulation in the fight for its protection is, in this author's view, nonsensical. Privacy advocates believe privacy to be an important part of our daily lives and deserving protection. After all, "anything worth protecting, is worth fighting for".

In the fight for privacy it would of course be naive to think, that any one means or technology could guarantee an individual's privacy. Rather, a combination of mechanisms, technologies and legislation are required. Such a combination will itself not be able to guarantee privacy, but would at least provide greater privacy protection, with definite consequences for those who invade or violate privacy, and those who fail to adequately protect the private information under their guardianship.

Perhaps the privacy mantra should become: *"Anything worth fighting for, is worth protecting"*.

Appendix A

Accompanying Material

The CD accompanying this dissertation contains the following:

- An academic paper titled "Towards Hippocratic Log Files", presented at the Information Security South Africa 2004 Conference in Midrand, South Africa, 30 June – 2 July 2004.
- An academic paper titled "Towards a Hipppocratic Log File Architecture", presented at the SAICSIT 2004 conference in Stellenbosch, South Africa, 4 – 6 October 2004.
- Files of the exploratory prototype discussed in chapter 7.
- A SQL server database backup file, which can be used to restore a copy of the database used with the exploratory prototype.

References

Abe, M. (1999). *Utilities for a Desert Isle.* Available from:http://www.stcsig.org/oi/hyperviews/archive/99Spring/992f1.htm. (Last cited:03 Jan 2005)

Agrawal, R., Kiernan, J., Srikant, R., & Xu, Y. (2002). *Hippocratic Databases.* Available from:citeseer.ist.psu.edu/agrawal02hippocratic.html. (Last cited:01/12/2004)

Ashley, P., Hada, S., Karjoth, G., Powers, C., & Schunter, M. (2003). *Enterprise Privacy Authorization Language (EPAL 1.2).* Available from:http://www.w3.org/Submission/2003/SUBM-EPAL-20031110/. (Last cited: 02 Jan 2005)

Ashley, P., Hada, S., Karjoth, G., & Schunter, M. (2003). E-P3P Privacy Policies and Privacy Authorization. In *Proceeding of the ACM workshop on Privacy in the Electronic Society* (pp. 103–109). ACM Press.

Bailey, D. (2000). *Log File Issues.* Available from:http://slis-two.lis.fsu.edu/~log/issues~1.htm. (Last cited:01 Apr 2004)

Bayardo, R. J., & Srikant, R. (2003). Technological Solutions for Protecting Privacy. *IEEE Computer*, *36*(9), 115–118.

Belgers, W. (1996). *Firewalls - An Introduction.* Available from:http://www.surfnet.nl/innovatie/desire1/deliver/WP5/D5-1.html. (Last cited:01 Apr 2004)

Berghel, H. (2001). Digital village: Caustic cookies. *Communications of the ACM*, *44*(5), 19–22.

Berghel, H. (2002). Hijacking the web. *Communications of the ACM*, *45*(4), 23–27.

Cavoukian, A. (1998). *Data Mining: Staking a Claim on Your Privacy.* Ontario, Canada: Information and Privacy Commissioner.

Clarke, R. (1996). *Identification, Anonymity and Pseudonymity in Consumer Transactions: A Vital Systems Design and Public Policy Issue.* Available from:`http://www.anu.edu.au/people/Roger.Clarke/DV/AnonPsPol.html`. (Last cited:03 Jan 2005)

Clarke, R. (1999). Internet Privacy Concerns Confirm the Case for Intervention. *Communications of the ACM*, *42*(2), 60–67.

Computing Research Association. (2003). *Four Grand Challenges in Trustworthy Computing.* Available from:`http://www.cra.org/Activities/grand.challenges/security/slides.pdf`. (Last cited:15 Apr 2004)

Cranor, L. (2002). *Web Privacy with P3P.* Cambridge: O'Reilly.

Cranor, L., & Garfinkel, S. (2002). *P3P: Privacy Primer.* Available from:`http://www.oreillynet.com/lpt/a/1554`. (Last cited: 19 Jul 2004)

Cranor, L. F. (1998). Putting it Together: Internet Privacy: A Public Concern. *netWorker*, *2*(3), 13–18.

Cranor, L. F. (1999). Internet Privacy. *Communications of the ACM*, *42*(2), 28–38.

Cranor, L. F. (2003). P3P: Making Privacy Policies More Useful. *IEEE Security & Privacy*, *1*(6), 50–55.

Curtin, M., & Ranum, M. (2000). *Firewalls FAQ.* Available from:`http://www.faqs.org/faqs/firewalls-faq/`. (Last cited:01 Apr 2004)

Eirinaki, M., & Vazirgiannis, M. (2003). Web Mining for Web Personalization. *ACM Transactions on Internet Technology (TOIT)*, *3*(1), 1–27.

ExactTrend Software. (2001). *Log Files: Frequently Asked Questions.* Available from:`http://www.exacttrend.com/weblogsuite/faq.html`. (Last cited:01/01/2005)

Froomkin, M. (2000). The Death of Privacy? *Stanford Law Review*, *52*, 1461–1543.

Goldschlag, D., Reed, M., & Syverson, P. (1999). Onion routing for anonymous and private internet connections. *Communications of the ACM (USA)*, *42*(2), 39–41.

Haynes, S. (2002). *Microsoft Computer Dictionary Fifth Edition.* Redmond, Washington: Microsoft Press.

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 (TOIT)*, *2*(4), 276–306.

Internet Marketing Engine. (2001). *How to read Web server log files.* Available from:`http://internetmarketingengine.com/how-to-read-server-log-files.htm`. (Last cited:01 Apr 2004)

Karjoth, G., Schunter, M., & Waidner, M. (2002). *The Platform for Enterprise Privacy Practices - Privacy Enabled Management of Customer Data.* Available from:`http://www.citeseer.ist.psu.edu/karjoth02platform.html`. (Last cited: 01 Jan 2005)

Kaufman, J. H., Edlund, S., Ford, D. A., & Powers, C. (2002). The Social Contract Core. In *Proceedings of the eleventh international conference on World Wide Web* (pp. 210–220). ACM Press.

Kerkhofs, J., Vanhoof, K., & Pannemans, D. (2001). *Web Usage Mining on Proxy Servers: A Case Study.* Available from:`http://www.informatik.uni-freiburg.de/~ml/ecmlpkdd/WS-Proceedings/w10/kerkhofs_vanhoof_pannemans.pdf`. (Last cited:01 Mar 2005)

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

LeFevre, K., Agrawal, R., Ercegovac, V., Ramakrishnan, R., Xu, Y., & DeWitt, D. J. (2004). *Limiting Disclosure in Hippocratic Databases.* Available from:`http://www.vldb.org/conf/2004/RS3P3.PDF`. (Last cited:01 Jan 2004)

Lin, D., & Loui, M. C. (1998). Taking the Byte Out of Cookies: Privacy, Consent, and the Web. In *Proceedings of the ethics and social impact component on shaping policy in the information age* (pp. 39–51). ACM Press.

Martin, D. (2003). *Web Bug FAQ.* Available from:`http://www.bugnosis.org/faq.html`. (Last cited:01 Jan 2004)

McManus, S. (2002). Protecting Your Privacy Online. *Internet Magazine.*

Mulvenna, M. D., Anand, S. S., & Buchner, A. G. (2000). Personalization on the Net Using Web Mining. *Communications of the ACM*, *43*(8), 122–125.

Nicolai Law Group P.C. (2001). *Technology Use Policies.* Available from:`http://www.niclawgrp.com/memos/200112.html`. (Last cited:01 Apr 2004)

OECD. (1998). *Implementing The OECD Privacy Guidelines in the Electronic Environment: Focus On the Internet. Report DSTI/ICCP/REG(97)6/FINAL, Directorate for Science, Technology and Industry Committee for Information, Computer and Communications Policy.* Available from:`http://www.oecd.org/dataoecd/33/43/2096272.pdf`: Organization for Economic Cooperation and Development. (Last cited:01 Jan 2004)

Olivier, M. S. (2003). Database Privacy. *SIGKDD Explorations*, *4*(2), 20–27.

P3PWriter. (2004). *P3P Privacy and Web Logs.* Available from:`http://www.p3pwriter.com/LRN_031.asp#1.0`. (Last cited:16 Nov 2004)

Presler-Marshall, M. (2000). *Web Privacy and the P3P Standard.* Available from:`http://www7.software.ibm.com/vad.nsf/Data/Document2363?OpenDocument&p=1&BCT=1Footer=1`. (Last cited:01 Jun 2003)

Rannenberg, K. (2004). Identity Management in Mobile Cellular Networks and Related Applications. *Information Security Technical Report*, *9*(1), 77–85.

Rao, J. R., & Rohatgi, P. (2000). Can pseudonymity really guarantee privacy? In *Proceedings of the Ninth USENIX Security Symposium* (pp. 85–96). USENIX.

Reagle, J., & Cranor, L. F. (1999). The Platform for Privacy Preferences. *Communications of the ACM*, *42*(2), 48–55.

Reiter, M. K., & Rubin, A. D. (1999). Anonymous Web Transactions with Crowds. *Commun. ACM*, *42*(2), 32–48.

Rezgul, A., Bouguettaya, A., & Eltoweissy, M. (2003). Privacy on the Web: Facts, Challenges, and Solutions. *IEEE Security & Privacy*, *1*(6), 40–49.

Rose, E. (2001). Balancing Internet Marketing Needs with Consumer Concerns: A Property Rights Framework. *ACM SIGCAS Computers and Society*, *31*(1), 17–21.

Sarma, S., & Mohirikar, N. (2003). *Logs and Forensics.* Available from:`http://www.cert-in.org.in/presentation/Logs-Forensics.pdf`. (Last cited:01 Apr 2004)

Schunter, M., & Ashley, P. (2002). *The Platform for Enterprise Privacy Practices.* Available from:`http://www.semper.org/sirene/publ/AsSc_02.EP3PatISSE.pdf`. (Last cited: 02 Jan 2005)

Sit, E., & Fu, K. (2001). Inside Risks: Web Cookies: Not Just a Privacy Risk. *Communications of the ACM*, *44*(9), 120.

Softsteel Solutions. (2003). *The Platform for Privacy Preferences Project (P3P).* Available from:`http://www.softsteel.co.uk/tutorials/P3P/index.html`. (Last cited: 12 Nov 2004)

Somerville, L. (2002). *Seeking Security Within.* Available from:`http://triad.bizjournals.com/triad/stories/2002/07/22/focus1.html`. (Last cited:01 Apr 2004)

Surferbeware.com. (2003). *Spyware FAQs.* Available from:`http://spyware.surferbeware.com/spyware-faqs.htm`. (Last cited:28 Jan 2004)

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

Tavani, H. T., & Moor, J. H. (2001). Privacy Protection, Control of Information, and Privacy-Enhancing Technologies. *ACM SIGCAS Computers and Society*, *31*(1), 6–11.

Tec-Ed, Inc. (1999). *Assessing Web Site Usability from Server Log Files.* Available from:`http://www.teced.com/PDFs/whitepap.pdf`. (Last cited:01 Jan 2004)

Thomas, B. (2000). *Intrusion Detection Primer.* Available from:`http://www.linuxsecurity.com/feature_stories/feature_story-8.html`. (Last cited:01 Apr 2004)

Treese, W. (2000). Data Collection And Consumer Privacy. *netWorker*, *4*(4), 9–11.

Ullman, J. D. (1988). *Principles of Database and Knowledge-Base Systems, Vol. I.* Computer Science Press, Inc.

Wang, H., Lee, M. K. O., & Wang, C. (1998). Consumer Privacy Concerns About Internet Marketing. *Communications of the ACM*, *41*(3), 63–70.

Warren, S., & Brandeis, L. (1890). *The Right to Privacy.* Available from:`http://www.louisville.edu/library/law/brandeis/privacy.html`. (Last cited:28 Dec 2004)

Webopedia. (2004). *Spyware.* Available from:`http://www.webopedia.com/TERM/spyware.html`. (Last cited:28 Jan 2004)

World Wide Web Consortium. (2002a). *An Introduction to P3P.* Available from:`http://www.w3.org/P3P/introduction.html`. (Last cited: 01 Jun 2003)

World Wide Web Consortium. (2002b). *P3P and Privacy on the Web FAQ.* Available from:`http://www.w3.org/P3P/p3pfaq.html`. (Last cited: 12 Nov 2004)

World Wide Web Consortium. (2002c). *The Platform for Privacy Preferences 1.1 (P3P1.1) Specification.* Available from:`http://www.w3.org/TR/P3P11/`. (Last cited: 12 Nov 2004)