Towards a Hippocratic Log File Architecture

ANDREW RUTHERFORD AND REINHARDT BOTHA
Port Elizabeth Technikon
and
MARTIN OLIVIER
University of Pretoria

The World Wide Web (WWW) is fast becoming the central location for goods, services and information. The very factors that make the Internet such a powerful medium, combine to make the Internet a treasure trove of personal information regarding individual Web users. Users' movements and information are logged as they navigate the Web, often without their knowledge and definitely without their explicit consent. This has lead to internet users voicing concerns over the loss and violation of privacy. Inspired by the Hippocratic Oath, Agrawal et al. [2002] introduced the concept of Hippocratic database systems. These systems are responsible for the privacy of data they manage, and must comply with ten defined principles. In a previous paper, the authors investigated the feasibility of applying the ten principles of Hippocratic databases to the management of log files. The focus of this paper is to expand on the originally proposed Hippocratic log file architecture by introducing a layered view of the architecture. In examining this layered view, the major processes that would be involved in the implementation of Hippocratic log files will be discussed.

Categories and Subject Descriptors: D.4.6 [**Operating Systems**]: Security and Protection—*Access Controls*; *Authentication*; *Information Flow Controls*; K.4.4 [**Computers and Society**]: Electronic Commerce; K.6.5 [**Management of Computing and Information Systems**]: Security and Protection—*Authentication*; *Physical security*; *Unauthorized access*

General Terms: Security

Additional Key Words and Phrases: Hippocratic Databases, Log files, Privacy

1. INTRODUCTION

"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]

Internet users privacy concerns are on the rise, to the extent that they have rated a loss of privacy as their number one Internet concern [Tavani 1999]. Privacy is lost on the Internet due to the use of various techniques and technologies, including cookies, web bugs, spyware and numerous log files. The logging of user information, is particularly pervasive. To make matters worse, users do not expressly provide the information. Users may therefore, be unaware that information collection is occurring. Even if they are aware of the collection, they may not be sure exactly what information is being collected.

Most commonly the following information is logged: the IP address that initiated a request; the date and time of the request; the requested object; the size of the requested object; the request status code; the size in bytes of the requested object; the referring URL; and the browser version and platform [Pitzek 2001]. Should a user enter a site via a search engine, search criteria used would also be logged [Fotheringham 2004]. Log information can be linked to individuals via cookies or possibly through online forms i.e.information gathered directly (forms), can be linked information gathered indirectly (logs) [Tavani 1999].

A user, in many cases, is logged not only by the site to which he is connecting, but also at the point at which his request originated. In many cases this would be an Internet Service Provider (ISP) or possibly a proxy server. At these points, information that is even more personally identifiable may be collected. This could include, for

Author Addresses:
A.J. Rutherford, Centre for Information Security Studies, Port Elizabeth Technikon, Private Bag x6011, Port Elizabeth, South Africa; arutherf@petech.ac.za.
R.A. Botha, Centre for Information Security Studies,Port Elizabeth Technikon, Private Bag x6011, Port Elizabeth, South Africa; rbotha@computer.org.
M. Olivier, University of Pretoria, Department of Computer Science, Pretoria, South Africa; molivier@cs.up.ac.za.

example, usernames.

In a paper titled, “Towards Hippocratic Log Files”, [Rutherford et al. 2004], the idea of applying the principles of Hippocratic databases [Agrawal et al. 2002] to log files was investigated. That paper discussed the extent to which each of the ten Hippocratic database principles could be applied to log files. The intent of that paper was not to provide specific answers or solutions, but rather to raise relevant issues. The paper concluded with a high level architecture which could conform to Hippocratic principles.

The goal of this paper is to extend the original architecture. Once again not to provide specific answers, but to raise issues of relevance that will need to be considered if Hippocratic log files are to become a reality.

The remainder of the paper is structured as follows. Section 2 provides a summary of the application of Hippocratic database principles to log files. Section 3 revisits the high level architecture introduced in an earlier paper. Section 4 introduces and discusses a layered view of the Hippocratic log file architecture. In so doing several procedural issues relating to Hippocratic log file implementation will be addressed. Section 5 highlights related work in the area of maintaining Internet anonymity. Section 6 concludes this paper with a brief summary.

2. HIPPOCRATIC LOG FILES: THE PRINCIPLES

This section summarizes the application of Hippocratic database principles to log files. Each principle as laid down by Agrawal et al. [2002] will appear in italics, with the word “database” substituted with words “log file”. Following the principle will be a short summary of the applicability of that principle to log files.

2.1 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. Various reasons for logging user information exist, associating these reasons with the personally identifiable information collected seems feasible.

2.2 Consent

The purposes associated with personal information shall have consent of the donor of the personal information. The collecting of information for security reasons supersedes the right of the individual to provide consent [Rutherford et al. 2004]. Users, however, should be afforded the opportunity to provide consent for other collection purposes.

2.3 Limited Collection

The personal information collected shall be limited to the minimum necessary for accomplishing the specified purpose. Collection of information for security reasons may result in the “minimum necessary” being as “much as possible”. However, it is important to stress the difference between the collection and the use of information.

2.4 Limited Use

The log file shall only permit queries that are consistent with the purposes for which the information has been collected. The principles of limited use and limited collection can operate harmoniously, provided collected information is used only for purposes for which the user has provided consent. The limited use principle will place certain requirements on the manner in which log files are accessed. As such, mechanisms need to be in place to ensure that only persons with required access rights gain access to log file information. The current practice, of storing log files as un-encrypted plain text, will not provide sufficient protection to assure limited use.

2.5 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. The principles of limited disclosure and limited use are closely related. Issues needing to be addressed by a Hippocratic log file architecture, raised in the previous section are of equal importance to the principle of limited disclosure.

During the course of a forensic investigation, it may be required to disclose personally identifiable user information to third parties. Ensuring that this information is used only for purposes agreed to by the user, once it leaves the protected environment, may not be technically possible. Such scenarios will require a combination of human trust and the law.

2.6 Limited Retention

Personal information shall be retained only as long as necessary for the purposes for which it has been collected. While it may not be possible to specify and exact time period, for which information is retained for security reasons, the period should be reasonable. There may be cases where information must be retained for extended periods of time, for example, statistical reasons. However, personally identifiable information can be removed by a process of aggregation or sanitization, while still maintaining statistical value.

2.7 Accuracy

Personal information stored in the log file shall be accurate and up-to-date. For the most part this principle is a non-issue. The logging of information is an automatic, machine driven process, and the same concerns of data accuracy, involving manual human data entry, do not apply. 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.

2.8 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.

2.9 Openness

A donor shall be able to access all information about the donor stored in the log file. The degree of openness to information is one that needs to be addressed. Giving users unrestricted access to their logged information, may provide network intruders the opportunity to cover their tracks. If all information is to be made available for inspection, then mechanisms must be in place to ensure that this information cannot be altered or deleted.

2.10 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. Since a log file provide an audit trail of what has transpired on a system or network, a further log file may be required to monitor access to that log file. This additional audit log could be referred to if ever questions of compliance to Hippocratic principles are raised.

Based on the summary provided in this section, it is clear that the Hippocratic principles can, to a large extent, be applied to log files. The next section introduces a process view of a proposed Hippocratic log file architecture.

3. HIPPOCRATIC LOG FILES: A HIGH LEVEL ARCHITECTURE

Figure 1 represents a high level overview of a possible implementation of Hippocratic log files - designed to conform to the principles of Hippocratic logs. Much like the original Hippocratic strawman design, this architecture is aimed at raising relevant issues rather than provide a final design.

Figure 1 shows the major components involved in the proposed architecture, as well as the interaction and actions that take place between these components. A solid line indicates actions that will occur. A dashed line on the diagram 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 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. At this point users have the opportunity to terminate communication. Due to the temporary nature of the logs on this server, any information users released will discarded after a limited time. This ensures compliance to the Hippocratic principle of consent.

A user may also be logged when initiating a request 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.

Users, choosing to proceed (Figure 1(B)), may set up their privacy preferences. These users will at this stage be informed of all the purposes for which the site is collecting information. They can then choose whether they agree to the use of their information, for each collection purpose. This again enforces the principle of consent. To avoid the scenario of frequent users having to re-enter preferences, cookies might be used to recognize returning visitors. In such a case a user can be directly routed to the logged server - indicated by (C) in Figure 1. Users should, however, always have the option of changing their preferences.

(C) and (D) in Figure 1 indicate a user being routed to the main server. By this time a user has agreed to the logging of information, and has set up his privacy preferences. All activity occurring on the logged server is recorded to the log file (Figure 1(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 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 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 is to allow them access to the audit log only. In this way they may not see what information is stored, but will be able to see that their information was accessed, and for what purpose.

Questions of openness aside, all requests, as indicated by (F), 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 will 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). 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 ever questions of compliance to Hippocratic principles are raised.

Section four that follows, will introduce and discuss a layered view of the Hippocratic architecture.

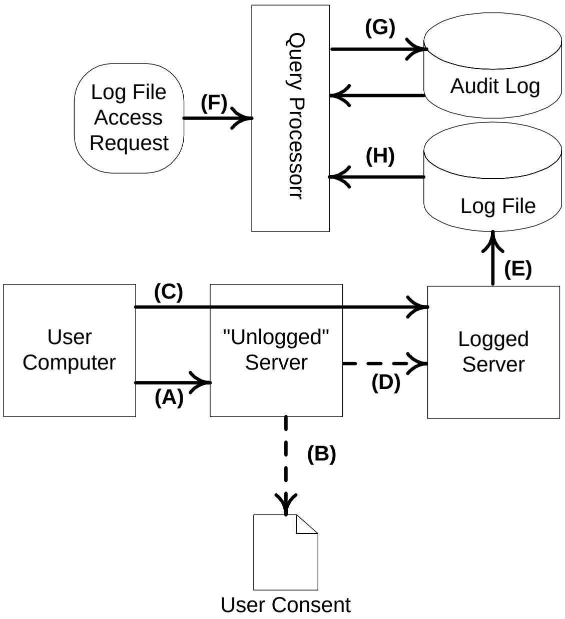

Figure 1. High-level Hippocratic Log File Architecture

4. HIPPOCRATIC LOG FILES: A LAYERED ARCHITECTURAL VIEW

The opening quotation in this paper, introduced the notion of providing users with a greater degree of control over their privacy, using security measures they can understand. A Hippocratic log file architecture should strive to provide users with just such control and security. The architecture should shield users from the 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, and the purposes for which this information is stored. Additionally they need to be provided with a user friendly, easy-to-understand means of consenting to the purposes of information collection. Allowing users to provide consent to information collection purposes, puts them in control of the further use of this information. 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 etc., can be well and truly hidden from users.

Figure 2 shows a layered view of the Hippocratic log file architecture. The architecture has been abstracted to three layers. The first layer are 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, as well as users consent choices, with regards to these storage purposes. The details of how this metadata will be stored and formatted, be it in XML, database tables etc., falls beyond the scope of this paper. Log files are accessed either for information retrieval or information storage. Surrounding log files with a layer of metadata is a 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, 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 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].

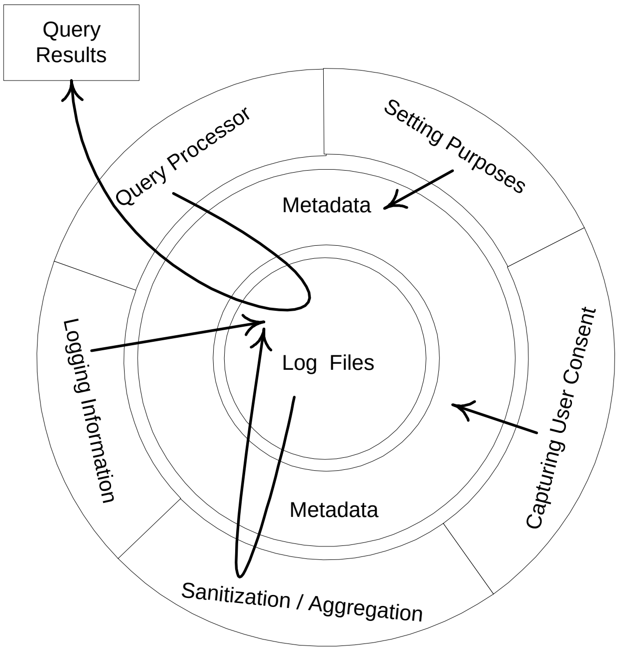

Figure 2. A Layered View of the Hippocratic Log File Architecture

4.1 Setting Up Purpose Metadata

Setting up purpose metadata would be the first step in establishing Hippocratic log files. This process is depicted by Figure 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. A list of users who are allowed access to this information should also be identified. This list can be used to ensure that only authorized users gain access 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, falls beyond the scope of this paper.

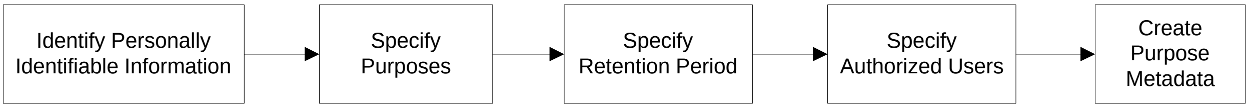

Figure 3. Setting Purpose Metadata

4.2 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. Users consent choices will be stored as consent metadata which is central to the proposed layered architecture. Figure 4 depicts the major steps involved in acquiring user consent.

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 is to provide a point at which users can view the reasons for information collection. For security reasons, this server will itself need to log information. However, users can be assured 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 be allowed however, 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 to capture employee consent metadata.

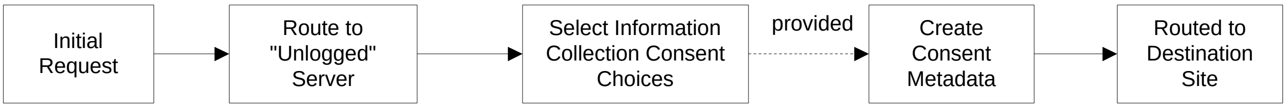
Initial Request
Route to "Unlogged" Server
Select Information Collection Consent Choices
provided
Create Consent Metadata
Routed to Destination Site

Figure 4. Creating User Metadata

4.3 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 as result in a great deal of purpose information repetition.

A further alternative might be to follow the approach as indicated in Figure 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 must be stored.

4.4 The Query Processor

The query processor will play a crucial role in enforcing that users information is used only for the purposes to which they have consented. Figure 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 that are for security related reasons, 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 reasons. 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 provided consent for this particular purpose of information usage, be returned. Each successful log file access will itself be logged for audit purposes.

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 to decrypt returned log file information.

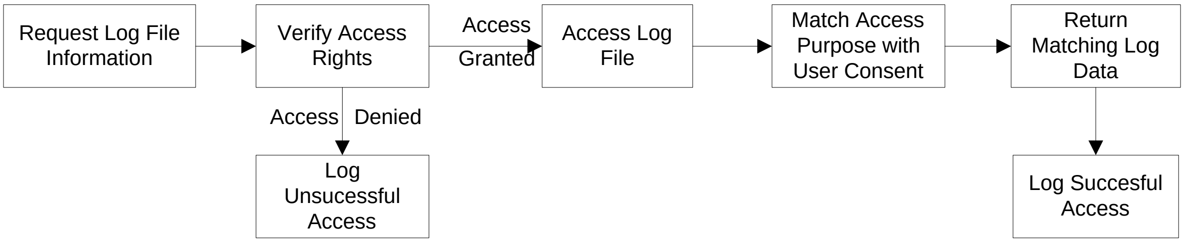

Figure 5. Log Query Analysis

4.5 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 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.

5. RELATED WORK

Various tools and technologies have been developed, in an attempt to limit the the logging of personally identifiable information. Such tools and technologies can provide users with a degree of anonymity when transacting on the Internet.

Users may subscribe to the services of an anonymous proxy, such as anonymizer [Cranor 1998]. This service allows all users' HTTP requests to be routed to a proxy based anonymizer, before submission to the destination site. Thus, with the proxy acting as a middleman as it were, no personally identifiable user information is received by the contacted Web site.

Private routing protocols, for example crowds, can also be used as a means to remove personally identifiable information. Crowds operates on the premise that “people can be anonymous when they blend into a crowd” [Cranor 1998]. All of a user's requests are forwarded through the crowd. When a member of the crowd receives the request, they can either submit it to the destination server, or to another randomly selected member of the crowd. By the time the 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 and Rubin 1999; Cranor 1998]. In this manner anonymity is assured.

These anonymizing tools suffer drawbacks that can be solved by Hippocratic logs. Firstly, A user may need to log into a destination server - in such a case the user may be identified, even if they arrived at the site via an anonymous proxy. Secondly, users subscribing to an anonymous proxy will be required to trust that this proxy will itself protect their private information. Thirdly, anonymous proxies will be unable to prevent a user's Internet service provider from monitoring and logging their Internet activity [Cranor 1998]. A further drawback of privacy enhancing technologies is that all too often they place the responsibility of preserving privacy and anonymity squarely on the doorstep of users.

6. CONCLUSION

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. These concerns can be alleviated, by providing users with a greater degree of control over their private information, with security measures they can understand.

This paper revisited the notion of applying Hippocratic database principles to log files, as well a high-level architecture originally proposed in a previous paper. A layered view of a Hippocratic log file architecture was introduced, in which log files are "surrounded" by metadata. This metadata stores the purposes for which information is stored, as well as user consent choices with regards to these purposes. All attempts to access log file information must pass through this metadata layer - thus ensuring that information is accessed according to users' consent choices. The proposed architecture can shield users from technical implementation details, while still giving them a greater degree of control over their private information.

Future work will include investigating means to automate the harmonization of users' privacy choices, with the information collection purposes of companies, or Web sites, with which they transact. This investigation will consider the use of E-P3P, to assist the automation process. E-P3P is a privacy policy model, that enables companies to define the manner in which they will use collected information. More importantly, E-P3P provides mechanisms to enforce a company's defined privacy practice [Ashley et al. 2003].

ACKNOWLEDGMENTS

The financial assistance of the National Research Foundation (NRF) towards this research is hereby acknowledged. National Research Foundation Grant number 2054024 (Olivier) and Grant number 2050802 (Botha and Rutherford).Opinions expressed and conclusions arrived at, are those of the authors and are not necessarily the to be attributed to the National Research Foundation.

REFERENCES

Agrawal, R., Kiernan, J., Srikant, R., and Xu, Y. 2002. *Hippocratic Databases.* Available from:http://www.almaden.ibm.com/software/dm/Hippocratic_Databases/index.shtml. Last cited:01 Apr 2004.

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

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. F. 1998. Putting it together: Internet privacy: a public concern. *netWorker 2,* 3, 13–18.

Fotheringham, J. 2004. *A Web server log file sample explained.* Available from:http://www.jafsoft.com/searchengines/log_sample.html. Last cited:01 Apr 2004.

Pitzek, S. 2001. *Security - Privacy on the Internet.* Available from:http://www.vmars.tuwien.ac.at/courses/akti12/journal/01ws/article_01ws_Pitzek.pdf. Last cited:01 Apr 2004.

Reiter, M. K. and Rubin, A. D. 1999. Anonymous Web transactions with Crowds. *Commun. ACM 42,* 2, 32–48.

Rutherford, A., Botha, R., and Olivier, M. 2004. Towards Hippocratic Log Files. *submitted.*

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

A. Rutherford, R. A. Botha, and M. S. Olivier, "Towards a Hippocratic log file architecture," in *Proceedings of SAICSIT 2004 — fulfilling the promise of ICT*, G. Marsden, P. Kotzé, and A. Adesina-Ojo (eds.), Stellenbosch, South Africa, 186–193, October 2004.