29
Analysis of Free Haven anonymous storage and publication system

Drake Patrick Mirembe, Francis Otto

In this paper we evaluate the design of a distributed anonymous storage and publication system that is meant to resist even attacks of most powerful adversaries like the government. We present a discussion of whether the assumptions held in the design of Free Haven System (FHS) are achievable in the real world and the potential implementation handles. We point out the different possible attacks and voice our opinions on the suggested solutions. We end with recommendations on how to improve the FHS design and offer direction for future research efforts in distributed anonymous storage services.

Background

The need to share information anonymously has been in existence since time immemorial. According to Anderson [2], in medieval times, knowledge was guarded for the power it gave those who possess it. He noted that, for example, the bible was controlled by the church for the knowledge it gave those who possessed it. Besides being encoded in Latin, bibles were often locked up in what we would call highly restricted areas. However, the spread of technical know-how destroyed the guild that had accreted abuses over centuries of religious monopoly. Today, many oppressed individuals desire to publish their criticisms anonymously in order to avoid persecution from those in authority. It is situations like those mentioned above that have inspired the design and deployment of anonymous storage and publishing services or networks such as the Eternity service, Mojo Nation, Gnutella, Napster and among others. The events that befell Napster and Gnutella in 1999 and 2000 respectively [1] and the work of Anderson [2] provided extra motivation for the development of a new generation of anonymous publishing service such as FreeNet and Free Haven System (FHS). As one can imagine the threat model of such a system would include most powerful adversaries such as "the governments and terrorists".

A. FHS Overview

The FHS concept is based on the use of a network of servers called ServNet; each server in the community holds segments of some documents referred to as shares which are created from documents by use of Rabins Information dispersal algorithm [4] and are traded between servers based on a buddy system [1]. To introduce accountability in the system, FHS uses a reputation system, which is

based on the performance of a server in transactions within the ServNet instead of a more complex digital cash system [5] as proposed by Anderson. To achieve server anonymity, servers are referred to by their pseudonym in the network. FHS relies on the secure MixNet for communication between ServNet nodes and the buddy system to check corrupt nodes on the network. To retrieve a document, the reader generates a key pair (PKclient, SKClient) and broadcasts the hash of the document together with a one time remailer reply block. A node that has a share of the requested document will reply the request using the PKClient and the remailer reply block information, and after receiving sufficient shares, the client reconstructs the document. In all, the FHS design emphasizes distributed, reliable, anonymous storage and publication over efficient information retrieval [8].

While in systems like FreeNet a publisher refers to a server, in the context of FHS, a publisher is an entity that places document in the system, while an author is the entity that originally created a document.

B. Paper structure

The paper begins with a brief background on the subject of anonymous publication and FHS in Section One. We present an overview on the design of FHS and a discussion on the subsystems that make up FHS which include the publication system, the reputation system and the communication system in Section Two. We proceed with an enumeration of possible attacks on system in Section Three, which may be social, political, technical and legal attacks. We present a discussion about the successes and failures of FHS in Section Four and we end with recommendations and future research directions in Section Five

FHS Design and Operation

The design of free haven was partly inspired by the earlier work of Anderson [2] when he proposed the Eternity anonymous storage service in 1996. Other projects, like FreeNet [3], Mojo Nation, Gnutella, Publius and Napster have highlighted the need to design a more robust anonymous storage and publication system. The design of FHS consists of mainly two parts, the publication system, which is responsible for storing and serving documents and the communications systems which provides a channel for anonymous communication between servers [1]. Little is added on the communication system as FHS basically uses the remailer network concepts which are already deployed in Mixmaster remailer networks including onion routing, freedom network and cypherpunk [1]. To improve the performance of FHS, the design allows publishers to set share expiry date, which guarantee that even if the document is unpopular, it will stay in the system as long as desired by the publisher. This is one of the improvements FHS added to the concept of anonymous publishing adopted earlier by FreeNet, Mojo Nation among others. In the following sections we give a detailed description of goals, design, and operation of FHS.

A. Goals of Free haven Project

Most of the early anonymous systems aimed at providing storage anonymity but not publication anonymity which led to the events that befall Napster and Gnuttela. In order to come up with a true anonymous storage and publishing system, the FHS team set the following as their design goals.

Provide **author anonymity;** according to [1] the FHS is designed to offer author anonymity. A system is said to be author anonymous if an adversary cannot link an author of document to the document itself.

Document anonymity; this means the contents of the document can not even be read by the server that is storing the document. For the survival of FHS, this property is crucial, as possession of an illegal content in a given jurisdiction can be a cause of censor to the server operator.

Publisher anonymity; this property is important in order to prevent an adversary from linking a publisher of a given document to the document itself.

FHS seeks to provide **Server anonymity;** this property means, given document identifiers, an adversary is not closer enough to link a document to the server(s) where it is stored.

FHS is **reader anonymity**; in order to enhance the privacy of the readers, FHS is designed with a module that makes its hard for an adversary to link a reader of a document to the document itself.

Query anonymity; by using the concepts of private information retrieval [6], FHS aims at preventing a server from determining which document it is serving to answer a user request.

Longevity of unpopular documents; The FHS system is designed with a mechanism that allows documents to remain in the system as long as the publisher desires. The duration of a document depends on the publisher of the document but not on the popularity of the documents itself as the case with FreeNet and others.

B. The Publication System

The FHS publication system uses the concept of (n, k) secret sharing schemes. A good example of these schemes is a polynomial of degree n $P = f(x)$, $f(x) = a_n x^n + a_{n-1} + x^{n-1} + \ldots + a x + a$ for where by only k out of n points are sufficient to evaluate. Thus, for an author to publish a document in FHS, he first looks for a ServNet node willing to store his document D, then applies Rabins information dispersal algorithm (RID) [4] on the document to break it into n shares of which any k out of n shares are sufficient to reconstruct the document. The publishing server then generates a public-secret key pair (PK_{doc}, SK_{doc})(axaxaxaxfnnnn++++==−−)(xfdoc, SKdoc), constructs and signs a data piece for each share (for accountability purposes during trading sessions) as well as insert other control information into the data piece like time stamp, expiry date for the share, share number and Pkdoc for verification purposes. For robustness, optimal k must be used, since a large k comparative to n makes the document D unrecoverable if a

few numbers of shares are corrupted while a small k means more storage space for a single share [3].

To provide cover for the publisher of a document and to allow servers to freely enter and leave the network without suffering damage to their reputation, servers periodically trade their shares within the ServNet. This provides a moving target to an attacker and therefore makes it difficult for any adversary to discover who the publisher of a given document is.

For secure and efficient retrieval, document shares in FHS are indexed by the hash of their public keys. And therefore, to retrieve a document, a searcher who is also a member of the ServNet generates the key pair (PKclient, SKclient) and one time remailer reply block for this transaction. Then, the searcher server broadcasts a document request H (PK_{doc}) along with its client public and reply block to its buddies. Any server that has a share of a document with H (PK_{doc}) as its index replies the request by encrypting the index of the share using the client's public key and forwarding it to the address provided in the reply block. After receiving at least **k** shares out n of **D**, the client reconstruct the document [1, 3 and 12].

C. The Communication System

For anonymous communication, FHS relies on the existing anonymous communications schemes such as Mixmaster remailers, Onion routing, and freedom network among others. However, the current design of FHS uses Mixmaster remailers and cypherpunk as anonymous communication channels. These schemes use the chaining technique to send encrypted messages to other nodes via a chain of nodes which can not read the contents of the packets. Since little is added to these schemes in the design of FHS, the reader is therefore referred to [1, 2] for a detailed description.

D. The Reputation System

To prevent attacks engineered by some malicious servers within the ServNet and to improve the trust between nodes, FHS relies on the reputation system. Many opportunities exists for servers to be naughty, servers can delete shares before their expiry date, wrongfully accuse other nodes for deleting shares, refuse to send acknowledgements after transactions among others [1, 5 and 7]. The design of FHS incorporates a "buddy subsystem" that creates dynamic association of servers holding shares of a given document. In this scheme every server maintains the reputation and credibility values of other servers in the buddy list basing on the number of successful transactions a given server say (A) has made, complaints received from other server about A, reputation of new servers A has introduced, validity of complaints A broadcasts among others. This reputation gives a degree of confidence servers put in a given server in regard to its adherence to the free haven protocol and the value of control information that server generates. This careful monitoring of buddies allows servers to keep track of buddies to trust in the ServNet [1, 7].

Also the system provides an easy way to add new servers and removing inactive ones. As servers with good reputation scores are trusted to act as introducers of newcomers in the ServNet via anonymous communication channel. Hence, allowing the dynamic growth and shrinking of the ServNet [7].

Attacks on Free Haven

The threat model of Free Haven system comprises some of the most advanced adversaries; Governments, corporations, and individuals all have reasons to oppose the deployment of FHS. The reasons may include copy right protection, fight against child pornography, fight against terrorism among others. These adversaries may employ technical, legal, political and social schemes as means to undermine a successful deployment of the system. In the proceeding sections we elaborate on these attacks.

A. Technical attacks

Technical attacks come from individuals, corporations and national intelligence agencies, targeting either the system as a whole or particular documents or servers in a given location so as to reduce the quality of service or gain control of part of the network. Given the post 9/11 world we live in with diminishing privacy and anonymity, national security agencies will deploy all the resources at their disposal to again access to documents in the system since the deign of FHS may provide a secure communication channel for terrorists and criminals. By use of viruses, worms and spy ware, security agencies and individuals can join the network and collect vital user identifiable information hence; violating the privacy of the users. Beside, the worms and viruses may be used to shutdown sections of ServNet.

Adversaries may also flood the system with queries so as to use up available resources hence, the denial of service attack [1, 7]. Other attacks like buddy co-opting in which the adversaries may join the ServNet and try to gain control of a good number of nodes, simple betrayal, server refusal to issue receipts at the end of a trading session, share hoarding in which a server trades away enough garbage so as to gain control of a significant amount of shares of a targeted content, false referrals in which a server broadcasts false reputation scores of other servers or simply forwards scores to only selected collaborating servers, log publishing and traffic analysis are all possible [1, 9].

B. Political and Legal attacks

The most difficult attacks to defend against are those that are politically engineered; individuals in the positions of authority in a given jurisdiction can use their influence to discourage use of the system. The authorities can attempt to find a physical server containing controversial documents and order its operators to shut it down or even prosecute the owners.

In some cases ordinary citizens may also employ the power of the government through lawsuits, multinational corporations who feel threatened by the deployment of FHS as it can encourage corporate espionage and infringement of

copy rights could persuade countries in which they operate to pass laws that bar the deployment of anonymous storage and publishing networks such as FHS. The Motion Picture Association of American (MPAA) and its world wide counterpart Motion Picture Association (MPA) and the Recording Industry Association of America (RIAA) have demonstrated resilience in fighting P2P networks such as Napster and Gnuttela for their facilitation in the distribution of copy righted content. We strongly believe it is organization like MPA and RIAA that pose a serious threat to a successful deployment of FHS.

C. Social attacks

The degree of social attacks on anonymous systems depends on the culture of the society in which the system is deployed. Some cultures like African cultures associate privacy and anonymity with evil, since in their culture; evil acts are committed in secret places and thus, whoever demands privacy and anonymity is a suspect of evil doing. Therefore, operating FHS in such societies will always be difficult; citizens will try to influence their governments so as to undermine the trust in the security of the system, as well as question the moral justification of ServNet node operators. These attacks can take the form of demonstrations and campaigns against the deployment of FHS, black mailing the operator of ServNet nodes of inappropriate conduct in society like facilitating child pornography among others. The current social pressure exerted on other P2P networks like KaZaA, Limewire, Napster among others strengthen our claim that FHS will also be subjected to the same.

Evaluation

Given the fact that at the time of writing this paper, FHS was and (is) still largely a conceptual design though with a proof of concept implementation by Dingledine [1] and his team, to our knowledge little in literature is discussed about the success and failures of FHS as a system. Although, FHS shares most of its properties with other distributed anonymousing systems like FreeNet, its good to mention that it's design introduces a number of concepts in the study of anonymous publication and storage systems such as query anonymity, document anonymity, publisher defined life span among others and our evaluation in this section is entirely based on the design of the FHS, the validity of assumptions made and the literature about similar systems particularly FreeNet.

A. Weakness

One of the weaknesses of FHS is its communication infrastructure. FHS relies on the existing anonymous communication channels for linking nodes within the ServNet, but these communication schemes such as Mixmaster and Onion routing (e.g. TOR) are unreliable and inefficient and therefore suffer from number of attacks, such as;

Traffic analysis attacks [1, 7, 8, and 9]; Mixmasters can be subjected to traffic analysis attacks by either a local or a global observer who can monitor traffic in

the ServNet and basing on the statistics (such as bandwidth) he can deduce, the identity of the communicating parties (source or destination) hence the loss of publisher and reader anonymity which is one of the goals of FHS. FHS team has developed a new generation of Onion routers called TOR [7] which is hoped to be more resistant to traffic analysis, however TOR has a weak threat model and a low user base. Since its inception, TOR has only remained in the hands of enthusiastic privacy and anonymity researchers due to its failure to win general user confidence and thus it seems less promising than first thought. Technically given its weak threat model TOR, does not promise much in the prevention of traffic analysis.

Usability weakness; the current design of FHS assumes that users have global knowledge of the network (keep a database of all public keys of nodes in the ServNet and reputation of buddies), which is absolutely impractical in an ideal peer-to-peer network made up of hundreds of thousands of peers. Users often have heterogeneous computing resources which may not be adequate to allow such extra processing and the size and dynamism of the P2P system makes this concept even more remote to achieve.

The design specifies the use of a buddy system which creates an association of nodes based on shares the server is holding so as to achieve accountability [1, 7]. Since servers query one another so as to determine the buddies of shares, it mean servers in FHS can read share identifiers and can determine which server hold what document as long as they hold buddy shares of a given document. Once more, receipts exchanged during the trading sessions between servers reveal more share identifiable information such as share number, expiry date, size among others. Thus the design only offers partial document anonymity from external attacks not from collaborating ServNet nodes and as such, document anonymity is hard to achieve in the current circumstances.

The claim that passive-server document anonymity property of FHS can protect a ServNet node against legal action is unrealistic. No technological means can supercede the laws of the land even in highly democratized societies plausible denial of data stored on ones server can not easily be an excuse against wrong doing in society and thus prosecution and arrests and other form of social and legal harassments are still possible.

The current design has no realist protection against "share hoarding attack" [1, 7]. The assumption that a collaborating group of individual server operators will find it expensive to obtain a fraction of a document due to the size of the network can not be realistically accepted. Dedicated and well facilitated spy organizations like the central intelligence agency (CIA) and terrorist organization like Al Qaeda can find enough resources to fund and form such expensive group.

Server anonymity is hard to achieve against bad hosts in the system and bad guys can easy join the system as they is no robust protocol defined to evaluate the reputation of new members, the reliance on introducers is not enough as introducers themselves mighty be collaborating with bad guys. The design of FHS

promises a lot, but in the current environment, implementing such a design is not easy and some of the assumptions do not pass some credibility tests.

We note that, the current design of FHS does not support document revocation as possessing extra knowledge to allow document revocation makes one a target of physical and legal attacks [1, 3 and 7]. However, in some case document revocation may be a desired property.

B. Successes

In comparison with its predecessors such as Mojo Nation, Napster, Gnutella, and FreeNet, FHS has a number of recommendable successes in its design even at its infant stage. Some of the components of the system are built on strong logical concepts of its predecessors.

In comparison with FreeNet and Gnutella, FHS maintains a document in the system even if the document is unpopular; this is one of the contributions made by FHS team to the study of anonymous storage and publication systems [1, 7].

Unlike Mojo Nation and Napster which have some of their system services centralized (like content tracker and publication tracker in Mojo Nation and central indexer in Napster), FHS is a purely decentralized system which can not suffer from a single point of failure. Peers have both serve and client functionality and they rely on the reputation and accountability systems to monitor the integrity of peers and to find which peers to trust.

Based on its trading scheme and high number of ServNet node in different jurisdictions (assumption), then FHS offers more publisher anonymity then any of its predecessors and this has been the reason why KaZaA has remained put amidst legal pressures from MPA and RIAA. If the trades are frequent and the number of servers is in hundreds of thousands say, it's not easy for an adversary to assume that the server trading a given share is the publisher of the document. The trading also provides a moving target to would be adversaries, thus improving the confidentiality of share and document anonymity. The trading scheme also protect the system from malicious share flooding as a server can only send into the network as much information as it can store.

By using selective query flooding [1, 7 and 12], FHS performs better than most of its predecessors in network resource optimization. When retrieving a document, the reader randomly selects a server his knows and sends out his request. This improves the over all system perform as other channels are made available to other communicating nodes, which is not the case with say FreeNet, Eternity service and Mojo Nation which relies on query broadcasts.

Based on FHS perfect forward anonymity (a system is said to offer perfect forward anonymity if no identifiable information of participants remains after a transaction is complete). It works on the premise that no key used for the transfer of data may be used to derive any keys for future transmission. FHS achieves author anonymity because authors communicate with publishers via an anonymous channel and share contains no extra information about authors.

However this is only true if authors behave and they don't give extra identifiable information in the documents they write.

C. Suggestions

We suggest that shares of a given document should be stored based on a quota (threshold) scheme per server and per jurisdiction. A given server should not store beyond a given percentage of shares of a given document (say 15%) and at least every share must have one of its buddies in every jurisdiction.

Buddies should have a short life span and nodes should not be allowed to associate with the same nodes over and over again. This will decrease the chances of buddy co-opting by a malicious server from happening. Of course short buddy life span means more network reconfigurations and hence more computational resources. Therefore a compromise should be made between computational efficiency and system robustness.

Since servers in the system maintain share identifiable information of which shares they have traded away and received, we can exploit this important information to improving efficiency of document retrieval by adapting the Intelligence Search Algorithm (ISA) [10, 12] instead of the current random walk algorithm.

Conclusions and Future

In all fairness one would say that, the chances of success of such a complex anonymous publication and storage system are slim. The greatest challenge of all is the requirement of implementing strong anonymity and strong authentication. These two properties are like an egg and chicken puzzle. Making one strong leads to the other being weak besides, the current design of FHS is partially anonymous and preventing the adversary from obtaining user identifiable information like their bandwidth, geographical location of the place where they live and share locations is practically impossible. The type of threats the system faces are very complex and adversaries can deploy extensive techniques to undermine the success of the system.

However, on the bright side, the popularity of Napster and Gnuttala systems provides hope that if technical and social handles are minimized to acceptable levels, The Free Haven service can be a success since the potential of attracting a critical mass of users is there. We have to mention that FHS design introduced more concepts in the area of anonymous storage systems, like the enhancement of publisher anonymity, server anonymity, document anonymity, query anonymity and reputation system based on buddies [1, 7].

The possibility of attacks through compromised or malicious servers hasn't been considered well in the current design and therefore requires further understanding and many of the issues that apply to other anonymous storage service like FreeNet do apply to FHS. Therefore, more research is needed in the direction of improving the accountability and reputation systems without comprising the robustness

and anonymity of servers, document and users. Our consensus is that under the current assumptions the use of discretionary distribution of shares in FHS is an improvement to the design of distributed anonymous storage systems.

It is clear from the literature that FHS promises a lot, but implementing such a system in a real world is still a long way off, the core issue being the provision of anonymity. Guaranteeing anonymity is very hard since it largely depends on building trust along the communication channel which in FHS design the communication channel is public subjected to all sorts of attacks. The failure of its predecessors due to legal pressures and lack of general acceptability in the society cast doubts on the successful implementation of Free Haven. It is our view that Roger Dingledine and his team have to faces an uphill task of achieving their dream.

References

Roger D, J.F Michael and M David "The Free Haven Project; Distributed Anonymous storage Service", International workshop on desiging privacy enhancing technologies: design issues in anonymity and unobservability, Berkeley, CA, USA, pp 67-95, Springer-Verlag, NY

Ross J. Anderson "The Eternity Service". In the proceedings of Pragocrpt 1996, Cambridge University Computer Laboratory

Ian Clarke "the free network project." http://freenet.sourceforge.net accessed on September, 30th 2006

Michael O. Rabin, "Efficient dispersal of information for security, load balancing, faulty Tolerance", Journal of the ACM (JACM) Vol. 36 , Issue 2 (April 1989) pp 335 - 348

David Chaum "Security Without Identification: Transaction Systems to Make Big Brother Obsolete". Commun. ACM 28(10): 1030-1044 (1985)

Y.Gertner, Y. Ishai, E. Kushilevitz and T. Malkin, "Protecting Data Privacy in Private, Proceedings of the thirtieth annual ACM symposium on Theory, 1998

Information Retrieval Schemes" http://theory.lcs.mit.edu/~cis/pir.html accessed on July 20th, 2006

FreeNet systems, www.freehaven.net accessed on August, 2006

The onion routr, http://www.onion-router.net/Publications.html accessed on July 22, 2005

Jean-François Raymond: Traffic Analysis: Protocols, Attacks, Design Issues, and Open problems. Proc. Designing Privacy Enhancing Technologies: Workshop on Design Issues in Anonymity and Unobservability (2000): Vol. 2009, pp 10-29

Demetrios Zeinalipour-Yazti "Information Retrieval in Peer-to-Peer Systems" Masters Thesis , June 2003, citeseer.ist.psu.edu/article/zeinalipour-yazti03information.html, accessed, November 5th, 2006

F. Otto and S. Ouyang: Improving Search in Unstractured P2P Systems: Intelligent Walks (I-Walks). In proceeding of 7th IDEAL conference; Sept 2006, Burgos, Spain; LNCS 4224, pp 1312-1319, springer.