Edith Cowan University
Research Online

Australian Information Security Management Conference

Security Research Institute Conferences

2006

Network Forensics and Privacy Enhancing Technologies "living" together in harmony

Giannakis Antoniou
University of Melbourne

Stefanos Gritzalis
University of the Aegean

Originally published in the Proceedings of 4th Australian Information Security Management Conference, Edith Cowan University, Perth, Western Australia, 5th December, 2006
This Conference Proceeding is posted at Research Online.
http://ro.ecu.edu.au/ism/69

Network Forensics and Privacy Enhancing Technologies "living" together in harmony

Giannakis Antoniou* and Stefanos Gritzalis†
*Dept. of Computer Science and Software Engineering
University of Melbourne
giannos@iprimus.com.au
†Dept. of Information and Communication Systems Engineering,
University of the Aegean, Samos, 83200 Greece
sgritz@aegean.gr

Abstract

Privacy Enhancing Technology (PET) is the technology responsible to hide the identification of Internet users, whereas network forensics is a technology responsible to reveal the identification of Internet users who perform illegal actions through the Internet. The paper identifies the collision of these opposite-goal technologies and describes what happens in case they come across. Can a Network Forensics protocol discover the identification of an Internet user who is trying to be anonymous behind a PET? The paper also proposes a way to bridge and eliminate the gap between these two technologies.

Keywords

Network Forensics, Privacy Enhancing Technologies.

INTRODUCTION

The Internet is a spread virtual society where a number of different technologies exist to offer services to its citizens (Internet users). Among the number of available technologies, there are the Network Forensics and the Privacy Enhancing Technology (PET). These two technologies have almost the opposite goal. The PET is used by Internet users to hide their identity during their e-activities whereas Network Forensics is used by Internet users to discover the identity of others. There have been proposed a number of PETs (Gritzalis, 2004; Anonymizer, 2003; Reiter and Rubin, 1998; Dingledine et al, 2004; Shields and Levine, 2000; Golle and Juels, 2004; Moller et al, 2003; Rennhard and Plattner, 2004) and Network Forensics (Carrier and Shields, 2004; Shanmugasundaram et al, 2003) techniques. These technologies may come across each other during the communication between a client and a Server, where the client wants to have privacy (by using PET) while the Server wants to know the identity of that client (by using Network Forensics) in case the client may be an attacker.

The paper studies the behaviour of those opposite-goal technologies when they come across each other and makes suggestions on how these technologies can co-exist in harmony, fulfilling the requirements of both client and the Server. The paper is organized as follows: Section 2 describes the PET framework; Section 3 describes two Network Forensics protocols; Section 4 examines the efficiency of the Network Forensics protocols where a PET is involved; section 5 proposes a new approach in order to apply Network Forensics protocols without violating the privacy of the users and section 6 concludes the paper.

PET FRAMEWORK

Although there are many proposed PETs oriented in the communication anonymity (Pfitzmann and Hansen, 2006), all of them follow the same framework (figure 1). A user, who wants to become anonymous, sends the message to a Privacy Enhancing Entity (PEE). The PEE may forward the message to another PEE or send it directly to the Server. Before PEE forwards the message, it replaces the IP Address of the message with its own IP Address. By using this technique, the receiver of the message will not be able to identify the original sender of the message but the last PEE of the PET. Some PET protocols such as the TOR (Dingledine et al, 2004) and the Crowds (Reiter and Rubin, 1998), also offer message confidentiality between the user and the first PEE.

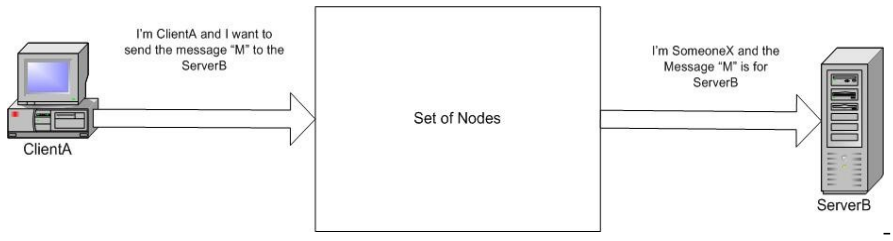

Figure 1 General PET Framework

NETWORK FORENSICS

In contrast with PETs, the Network Forensic techniques have different frameworks. In this section we describe two network forensics protocols, the STOP (Carrier and Shields, 2004) and the ForNet (Shanmugasundaram et al, 2003) protocols.

STOP protocol

The paper (Carrier and Shields, 2004) argues that an attacker usually does not directly attack a Server, but through a number of other linked compromised machines. The attacker usually follows this method in order to hide his/her identity. The paper proposes the Session Token Protocol (STOP), which can trace back the attacker even if he/she attacks through other compromised machines (Stone Stepping attack).

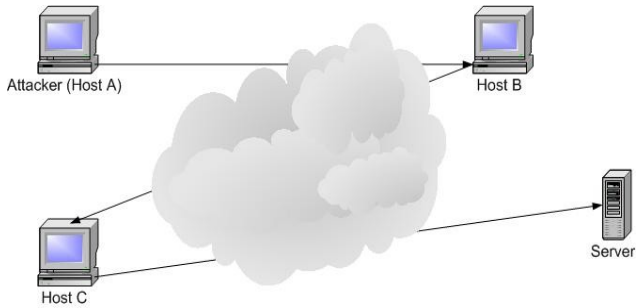

Figure 2 Stone Stepping attack

For example (figure 2), an attacker (hostA) wants to attack a Server by using a Stone Stepping attack (Zhang and Paxson, 2000) without being identified. Firstly, the attacker compromises the hostB, and then from the hostB compromises the hostC. The attacker attacks from hostC the Server while the Server can only identify the hostC. No information exists which links the hostC with the hostB and the hostB with the hostA (the actual attacker). However, by using the STOP protocol, the hostB and the hostC gather information about any network connection they establish, and under certain conditions they can inform the Server about the identity of the attacker. One drawback of this, is the fact that if the hostA compromises the hostB and then the hostC, the hostA can also delete the logs from hostB, as well as the logs from hostC. As a result, the Server will not be able to identify the attacker. Another drawback is the ability of the attacker (hostA) to accuse an innocent host (let say hostZ).

ForNet protocol

The paper (Shanmugasundaram et al, 2003) introduces a logging mechanism, called ForNet, which helps to identify the source network - not the source host - of a malicious packet in a distributing and sharing environment, like the Internet. The ForNet has two components, the Synopsis Appliance (SynApps) and the Forensic Server. The SynApps is located in every router/switch and records the synopsis of the transferred data (including information from the 3rd, 4th and 7th OSI Layer). The Forensic Server is responsible to manage the SynApps that are part of the specific domain, as well as help with tracking of a potential attacker. However, it concerns the performance of the routers/switches in case they manage information up to the 7th OSI Layer.

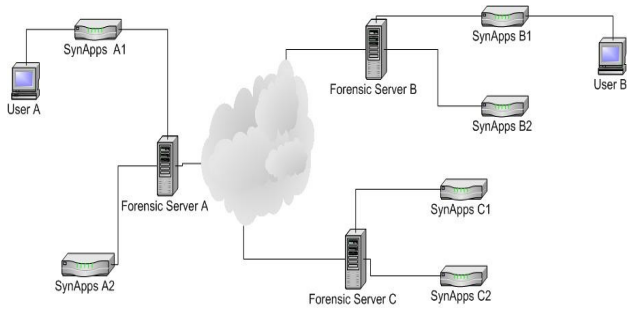

Figure 3 ForNet Architecture

Example

In figure 3, the UserA wants to send a malicious message to the UserB. The UserA passes the message to SynApps A1, where the SynApps A1 creates and stores the digest of the message before forwarding it to the Forensic Server A. The Forensic Server A forwards the message to the Forensic Server B and the Forensic Server B forwards it to the SynApps B1. The SynApps B1 creates and stores the digest of the message before reforwarding the message to the UserB. Once the UserB identifies the attack, it informs the SynApps B1 about the incident and the SynApps B1 requests the Forensic Server B to identify the source network of the message. The Forensic Server B sends a request to all Forensic Servers. Each Forensic Server asks its SynApps whether or not they have forwarded such a message digest. The SynApps A1 identifies the message digest and informs the Forensic Server A. Finally, the Forensic Server A lets Forensic Server B to know that the malicious message has passed from SynApps A1 at a specific instance of time (timestamp). In the meantime, more Forensic Servers may reply. However, the Forensic Server B will determine the original source network of the malicious message based on that timestamp. The earlier the message passes from a SynApp, the more probable that SynApp is the network, in which the UserA (attacker) belongs to.

Both Network Forensics protocols are able to trace back and identify the attacker/network source. However, the next section examines how they response in an environment with PETs.

PET AND NETWORK FORENSICS IN THE SAME ENVIRONMENT

This section examines the efficiency of STOP and ForNet protocols where the attacker sends the messages through the PET. An effective Network Forensics protocol should be able to identify the attacker who is even hidden behind a PET, while an effective PET should be able to protect the identity of a user.

STOP Protocol with PET

Although the STOP protocol is the answer of the Stone Stepping attack, it cannot identify the attacker in case that the Attacker is behind a PET.

In the example of figure 4, the attacker sends a malicious message to the PET asking the PET to forward it to the HostB. The PET forwards the received message to the HostB. Once the HostB receives the message, it stores information about the connection (PET - HostB) to a log file. After the HostB becomes compromised, the HostB (under the control of the Attacker) attacks HostC. The HostC stores information about the connection (HostB-HostC) and then becomes compromised. The HostC (still under the control of the Attacker) attacks the Server. However, the PET does not follow the STOP protocol; otherwise the PET should reveal the identity of the attacker. Consequently, the Server will not be able to discover the malicious host (Host A).

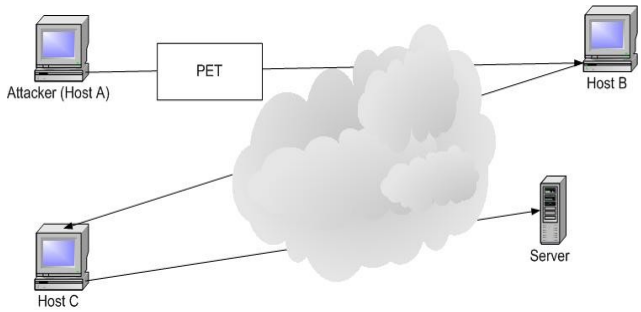

Figure 4 PET against STOP protocol under Stepping Stone attack

ForNet Protocol with PET

In a scenario (figure 5) where the attacker participates in an environment where the ForNet protocol is supported, the attacker wants to take advantage of the PET to hide his/her identity while is performing an illegal action. When the Server detects an attack, it will try to recognize the network source of the attack in assistance with both SynApps and Forensic Servers. Because of the involvement of the PET, who is responsible to hide the identity of the user, the Server will not be able to discover the network source of the attack.

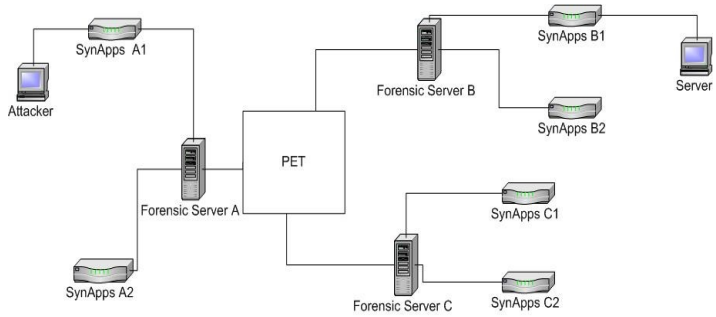

Figure 5 ForNet protocol with PET

The ForNet and the STOP protocols, like most of the Network Forensics protocols, use information from the 3rd and 4th OSI Layers to track the original source of a message. It is, therefore, impossible to be used in an environment where there is a PET which offers communication anonymity to the sender, because the PET modifies information from the 1 - 4 OSI layers.

We want to prove that the location of the Network Forensics entities is not able to reveal the identity of the attacker, when the attacker uses a PET. One of the main characteristics of the current PETs is their ability to prevent privacy violation from a global eavesdropper. As a consequence, the use of Network Forensics protocols is useless without concerning the functionalities of PETs.

Case studies

In the next case studies, the Attacker tries to attack the Server through a PET. We also show how effective is the Forensic Entity (an entity which helps identifying an attacker) in helping the Server to discover the identity of the Attacker. The Forensic Entity needs to link the message received by the Server and the message received by it (Forensic Entity). Otherwise, the Forensic Entity does not identify the attacker. We assume that the Forensic Entity eavesdrops the communication between the Attacker and the PET, or it is just an intermediate node between them.

We assume that the Server is able to identify the Attacker based only from the source IP Address of the malicious message.

Case Study 1

Let us assume that the communication between the Attacker and the PET is not encrypted

Let X: information from the 3rd and 4th OSI Layer

Attacker ➔ PET: X

PET ➔ Server: X'

The Server cannot identify the Attacker because the PET replaces the X with X' before forwarding the message to the Server.

Attacker ➔ Forensic Entity: X

Forensic Entity ➔ PET: X

PET ➔ Server: X'

In this case the Server cannot identify the attacker even with the co-operation of the Forensic Entity. The X that was sent by the attacker to the PET through Forensic Entity, does not match the X' received by the Server.

Case Study 2

Let us assume that the communication between the Attacker and the PET offers message confidentiality

Let X: information from the 3rd and 4th OSI Layer

Attacker ➔ PET: X, cipher

PET ➔ Server: X', message

The Server is unable to identify the Attacker.

Attacker ➔ Forensic Entity: X, cipher

Forensic Entity ➔ PET: X, cipher

PET ➔ Server: X', message

The Forensic Entity cannot match the [X, cipher] and [X', message].

Case Study 3

Let us assume that the communication between the Attacker and the PET is not encrypted

Let X: information from the 3rd and 4th OSI Layer

Let Y: information from the 7th OSI Layer

Attacker ➔ PET: X, Y

PET ➔ Server: X', Y

Although PET does not modify information from the 7th OSI Layer, the Server cannot identify the Attacker.

Attacker ➔ Forensic Entity: X, Y

Forensic Entity ➔ PET: X, Y

PET ➔ Server: X', Y

If the Forensic Entity collects information up to the 7th OSI Layer, it is able to match the information received by the Server with its own collected information. But, if the Forensic Entity collects information up to the 4th OSI Layer, it is unable to match the information and discover the Attacker (case study 1).

It is obvious that the PET causes problems to the current Network Forensics protocols once these two technologies come across. Only, when the PET does not offer message confidentiality between the attacker and the PET, the Network Forensics protocols are able to find the attacker. However, even in this case, the Network Forensics protocols have to gather information based on the 7th OSI Layer protocols. It is also clear that the

Network Forensics protocols, which gather information up to the 4th OSI Layer, are unable to help in the investigation.

In (Luis von et al, 2006) the dangerous nature of anonymous communication is mentioned, where the authors suggest ways in offering accountability to the anonymity protocols. The next section proposes a new approach in order to achieve accountability by combining the PET with the Network Forensics.

NEW NETWORK FORENSICS APPROACH

In order to make these two technologies avoid collisions and offer their services in an efficient manner to the Internet users, we need technologies which are able to separate the innocent users from the attackers. The PET can protect the identity of an innocent user whereas the Network Forensics can identify the attacker. This separation is a very sensitive operation, where the objectivity plays a fundamental role in the success of this approach. If we accept that this objectivity is not achievable, then this approach fails because the separation of the actual attacker from the innocent user is not possible.

The two network forensics phases

We can divide our abstract approach into 2 phases. During the first phase, each user has one warrantor. The warrantor authenticates the user and gathers information, without violating the user's privacy. The warrantor does not even need to know who the Server is, which the user is communicating with. Every user is considered as innocent to that warrantor, as far as the Server-victim does not complain. The Server accepts messages from an anonymous user only if there is a proof that the user is linked with a warrantor. In order to have that link between the user and the warrantor, the warrantor may sign (with digital signature) a statement. This signed statement obligates the warrantor to reveal the identity of the user in case the user is an abuser. This link must also be linked with the exchanged messages of the user. In case the user performs an attack, the Server can complain to the warrantor and the warrantor is responsible to investigate the case.

The second phase begins when the Server complains to that warrantor. The warrantor must be able to decide whether the complaint is reasonable or not. However, the Server needs to prove to the warrantor that the specific warrantor is responsible to investigate the complaint and reveal the identity of the potential attacker. The Server's complaint should be in such a way that the warrantor should not be able to deny its involvement in the investigation process. Moreover, the Server needs to provide strong evidence to the warrantor about the integrity of the malicious action of the attacker. Once the warrantor accepts the complaint as a reasonable one, it must have a mechanism to match the complaint with the identity of the attacker.

The warrantor is responsible for the incident as far as it does not reveal the identity of the anonymous user. The warrantor needs to examine objectively the complaint of the Server before deciding whether to reveal the identity of the user or not. This is the most critical aspect for the success of this approach for two reasons:

a) The criteria of the warrantor may be stricter than the criteria of the Server. In this case, the Server will not be able to get the identification of the attacker.

b) The criteria of the warrantor and the Server are very loose. In this case, the warrantor may reveal the identity of an innocent user.

In order to define what is "reasonable" and what is not, the warrantor may sign and send the security policy to the Server through the user during the first phase. The Server can decide whether the warrantor fulfilled the security requirements of the Server or not. If the policy is not acceptable for the Server, the Server rejects the request of the user, because the warrantor does not fulfil the requirements of the Server. The warrantor does not only need to reveal the identity of the attacker, but it needs also to prove the involvement of the attacker in the malicious action. Otherwise, the warrantor may accuse an innocent user instead the real attacker.

Although the identification of an innocent user can be revealed in case the warrantor becomes compromised by an attacker, the attacker will not be able to accuse (lack of evidence) that innocent user. This disadvantage does not create significant concerns for the users, because the PETs have exactly the same vulnerability (in case the

PET becomes compromised, the identities of the users can be revealed). The worst thing that a malicious warrantor can perform to the user is to reveal the link between the warrantor and the user to the public. The warrantor does not even know the Server's identity that the user is communicated with. The user can also play the role of the warrantor. Even in such a case, the Server-victim can accuse the warrantor or the abuser, even if the entity is the same.

Applications of the proposed approach

We introduce two applications based on the proposed approach. In the first application (figure 6), the PET is the warrantor, whereas in the second application (figure 7), an individual entity is the warrantor:

a) The enhancing of network forensics functionality in a PET is a challenging procedure. A PET should have a mechanism responsible, under conditions, to reveal the identity of the user. However, this approach violates the principle of the PET, because instead of protecting the identification of the user, it is actually betraying the user. Despite this principle of violation, the PET continues to offer privacy only to the innocent user but not to the attacker. The separation of the innocent user from the actual attacker is more likely to be acceptable by all participated entities except from attackers themselves.

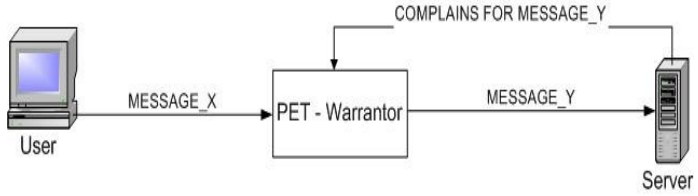

Figure 6 PET as a warrantor

b) Although the message will be sent by the user and forwarded to the Server through the PET, the warrantor should be able to recognize the user through the message that has been received from the Server. For this reason the message (MESSAGE_Y - figure 7) must:

   a. Be linked with the warrantor of the user. The Server needs to identify the warrantor before proceeding with the message.

   b. Hide the identity of the user. Responsible to protect the identity of the user is the PET.

   c. Offer message integrity. If the message does not offer integrity, the Server could alter the received message and make it look malicious. With lack of message integrity, the Server can state/assume to the warrantor that everyone is an attacker.

   d. Reveal the user's identity to the warrantor. The warrantor should be the only one who can match the message with the user's identity.

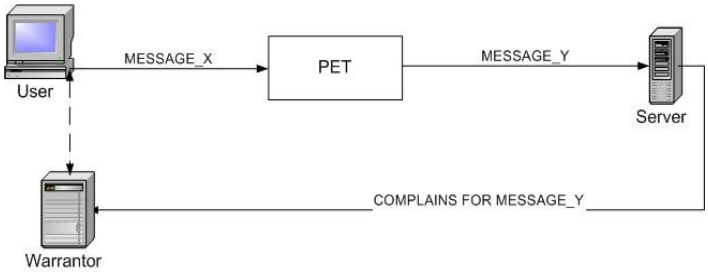

Figure 7 Individual Entity as a warrantor

The above applications offer network forensics services without any violation of the privacy of users, but only of the attackers. The user may enjoy the privacy offered by the PET, while the Server knows that in case of an attack the identification of the attacker will be revealed. The fact that the warrantor and the Server have evidence about the malicious action of the user, it can result in accusing the user.

CONCLUSION

This paper is oriented to the fact that two very important technologies, like the Network Forensics and the PET, cannot co-exist under the same environment and circumstances. The more one technology gets improved, the less the other technology becomes enhanced. We studied their behaviour in different case studies and we can conclude that the PET has really been designed having in mind technologies that their purpose was to reveal the identity of the users. In contrast, the Network Forensics protocols have not been designed having in mind technologies responsible to hide the identification of the users, like the PETs.

The paper proposes a new Network Forensics approach which is PET-friendly. Both opposite-goal technologies can "live" together in harmony offering their desirable services to the innocent users and the Servers.

Our future work is emphasizing in enforcing the objectivity of the warrantor in order to decide whether the complaint of a victim is reasonable or not. Also, a proper communication protocol, which reflects to the proposed approach, is required to be developed.

REFERENCES

Anonymizer (2003), URL *http://www.anonymizer.com*, Accessed at 16th of November 2006

Carrier, B. and Shields, C. (2004) The Session Token Protocol for Forensics and Traceback. *ACM Transactions on Information and System Security, Vol. 7, No. 3, August 2004, Pages 333-362*

Golle, P. and Juels, A. (2004) Parallel Mixing. *October 2004 Proceedings of the 11th ACM conference on*

Dingledine, R., Mathewson, N. and Syverson, P. (2004) Tor: The Second-Generation Onion Router. *In Proceedings of the 13th USENIX Security, Symposium, August 2004*

Gritzalis S. (2004) Enhancing Web Privacy and Anonymity in the Digital Era. Information Management and Computer Security, Vol.12, No.3, pp.255-288, 2004, Emerald

Luis von, A., Bortz, A., J. Hopper, N. and O'Neill, K. (2006) Selectively Traceable Anonymity. *Proceedings of the 2006 Workshop on Privacy Enhancing Technologies*

M¨oller, U., Cottrell, L., Palfrader, P. and Sassaman. L. (2003) Mixmaster Protocol. Version 2. Draft, July 2003, URL *http://www.abditum.com/mixmaster-spec.txt*, Accessed at 16th of November 2006

Pfitzmann, A. and Hansen, M. (2006) Anonymity, Unlinkability, Unobservability, Pseudonymity and Identity Management - A Consolidated Proposal for Terminology, *URL http://dud.inf.tu-dresden.de/literatur/Anon_Terminology_v0.28.doc*, Accessed at 16th of November 2006

Reiter M., Rubin A. (1998) Crowds: Anonymity for web transactions. ACM Transactions on Information and System Security (TISSEC), Vol. 1 , Issue 1 (Nov 1998), Pages: 66 - 92

Rennhard, M. and Plattner, B. (2004) Practical anonymity for the masses with morphmix. In Ari Juels, editor, Financial Cryptography. Springer-Verlag, LNCS 3110, 2004.

Shanmugasundaram, K., Savant, A., Br¨onnimann, H. and Memon, N. (2003) Fornet: A distributed forensics network. In The Second International Workshop on Mathematical Methods, Models and Architectures for Computer Networks Security, St. Petersburg, Russia, October 2003

Shields, C. and Neil Levine, B. (2000) A Protocol for Anonymous Communication Over the Internet. *November 2000 Proceedings of the 7th ACM conference on Computer and communications security Computer and communications security*

Zhang, Y. and Paxson, V. (2000) Detecting Stepping Stones. In Proceedings of the 9th USENIX Security Symposium, Pages 171-184, 2000

COPYRIGHT

Giannakis Antoniou & Stefanos Gritzalis ©2006. The author/s assign SCISSEC & Edith Cowan University a non-exclusive license to use this document for personal use provided that the article is used in full and this copyright statement is reproduced. The authors also grant a non-exclusive license to SCISSEC & ECU to publish this document in full in the Conference Proceedings. Such documents may be published on the World Wide Web, CD-ROM, in printed form, and on mirror sites on the World Wide Web. Any other usage is prohibited without the express permission of the authors