Countermeasures against traffic analysis for open networks

KAMIL KULESZA, ZBIGNIEW KOTULSKI
Institute of Fundamental Technological Research, Polish Academy of Sciences
ul.Świętokrzyska 21, 00-049, Warsaw Poland,
e-mail: <Kamil.Kulesza,Zbigniew.Kotulski>@ippt.gov.pl

Abstract:
In the paper we provide description of traffic analysis problem. We start from the real-life origins of this type of attack. Then we provide adversary models and assumptions on the network. Next, we outline few instances of possible attacks. Keeping high level of abstraction allows us to discuss traffic analysis prevention for various types of abstract networks and paradigms, for instance: packet networks, wireless environment, ad-hoc networks, real-time systems and mobile agent systems. We describe standard countermeasure (padding, routing) and discuss their limitations. Next, traffic analysis prevention metric is outlined. We summarize countermeasures with discussion of tradeoffs, which are inevitably associated with them. In addition, we discuss how excessive use of countermeasures against traffic analysis produce side channels, which will benefit the attacker.

Keywords: traffic analysis, security protocols, countermeasures, side channels

1. Introduction

Traffic analysis (TA) can be described as the process of monitoring rather nature and behavior of traffic, than its contents, see [1]. It is a very good example how concepts from the real-life migrate into cyberspace, making both realms to converge. This allows us to start TA's description from real-life examples and gradually move towards computer world.
TA roots can be traced back to human activity of passive observation. Passive observation seems to be as old as espionage itself, which claims to be the second oldest profession. The case of traffic analysis for intelligence agents was described in detail by Peter Wright in "Spycatcher: The Candid Autobiography of a Senior Intelligence Officer" [2]. In the book he describes how Russian agents were performing successful TA on British counterintelligence services in London during the Cold War. There are also accounts of the technical side of the story, TA depended heavily on monitoring radio transmissions between counterintelligence officers (which can be classified as SIGINT[1]).
In more computer related situations, TA originated from analysis of command and control in the military. For instance, state of alert of military command center is highly correlated with the patter in electronic traffic, see [3]. Encryption of the network packets causes only minor problem to an attacker, which performs traffic analysis. This is best illustrated by the following quote: *"By the act of communicating, even if perfect confidentiality of the actual information is achieved, one can give indications to observing parties of impending actions, capabilities, chains of command, and level of readiness."* (from "Research Challenges in High Confidence Networking", DARPA, July 1, 1998).
But TA is not only limited to the military systems, it can have far reaching consequences in the business environment. An excellent account of traffic analysis operation is given in "Wall Street", the movie directed by Oliver Stone (see [4]). Let us provide an outline of the plot, since it is essential for further discussion. The story is based on the famous Ivan Boesky case in 80', when a stock market tycoon was nailed by SEC (Securities and Exchange Commission) with charges of insider trading. In the picture stock market tycoon Gordon Gekko (Michael Douglas) sends his young apprentice Bud Fox (Charlie Sheen) to observe Sir Larry Wildman (Terrence Stamp). The later is a powerful British investor planning some deal in the US. Gekko wants to learn about the deal. Following Sir Larry Wildman for all the day Bud Fox finds what investment banks Brit was talking to. He also found where the investor flew his jet to after the talks. The apprentice was unable to get any detail of conversations, since these were carried out in secure locations. When Bud Fox came to report his master, he was apologizing for the poor results. But for Gordon Gekko it was enough information. Knowing the names of investment banks and people involved he was sure that Sir Larry Wildman was interested in some heavy industry enterprise. All these, combined with the Wildman's plane destination (some location in Pennsylvania) revealed his adversary the company itself—Anacott Steel. Such a reasoning was possible, since Gordon Gekko knew well Sir Larry Wildman. He combined his knowledge with all the information obtained by the traffic analysis. As a result he derived and implemented strategy that allowed him to make millions of dollars on the stocks market. Exactly the same scenario is applicable to case of electronic negotiations between companies, as mentioned in [5].

[1] SIGnals INTelligence

Above example can be described in more network related jargon as problem of protecting the identity of communication partners (or at least one of them). This can be also very much the case for an ordinary Internet surfer, which does not want third party to create his surfing dossier. This type of threat can range from just marketing profiling to tracking by some governments people viewing on the Web banned political information (e.g., CNN, BBC in some part of the world). Even when services hiding content and URL of the site visited, like SafeWeb (see [6]), are used, Internet surfer still might be vulnerable to traffic analysis (see [7]).
The paper organization is as follows: in the next section, we provide background information, while Section 3 is devoted to the network assumptions, adversary powers and possible attacks. In Section 4 we discuss countermeasures for various types of the networks and conclude in Section 5.

2. Background information and related concepts

From theoretical point of view, traffic analysis is closely related to:
- *network unobservability*, which is about hiding all communication patterns in the network;
- *unlinkablity*: message is not linkable to sender/recipient;
- *anonymity* (privacy protecting mechanisms) *& pseudonymity* (using pseudonyms as ID by senders/recipients).

While precise definitions for above terms are available (e.g., see [8]), for the sake of of clarity this paper their basic understanding should be sufficient. It should be also pointed out that applicability of some of these terms to traffic analysis is matter of ongoing dispute, for instance anonymity versus pseudonymity (see [9], [8]).
Below we list few concepts dealing with relation between sender and recipient:

a. secure multiparty computation, see [10];
b. private information retrieval, see [11];
c. Chaum's mix, see [12];
d. Chaum's DC-net, see [13];
e. Crowds, see [14];
f. onion routing, see [15].

Those form a toolbox that can help to prevent traffic analysis. We return to some of these concepts in Section 4. Now, it is time to see what adversaries and attacks are ahead.

3. Adversaries, Network Assumptions and Attacks

In the first part of this section we briefly describe assumptions that we make on the environment. Having that we are able to outline some of the most popular attack techniques.

3.1 Adversary and Network Assumptions

In general, in order to describe the attack one has to start from modelining of an adversary. This is usually done in terms of adversarial powers, for instance see [9]. Hence, the adversary can be:

a. *static or adaptive*. In the first case adversary makes a choice on what resources he compromises once and for all, before the protocols is started. In the latter adversary makes dynamical decisions, based on the information gathered during the protocol execution;
b. *local or global*. The difference is in what part of the system can be controlled by adversary;
c. *passive or active*. In the first case adversary can only observe the traffic. Some models (e.g., [16]) introduce global passive adversary, which is assumed to observe the entire message flows in the network. This is a very strong assumption, rather unlikely to happen in real-life. But active adversary is even worse: he can not only listen, but also remove messages or inject probing traffic into the victim network and observe the reaction, see [17];
d. *internal or external*. In first case adversary can compromise sender/recipient, while in second communication and possibly some of mix nodes.

In addition, an adversary has assigned computational power, which can be bounded (e.g. polynomial time) or unbounded. These have an impact on the adversary capabilities in terms of:

a. her abilities to compromise communication in the network (e.g., by breaking some crypto);
b. complexity of the analysis that she can perform.

Note, that adversarial models can be build with combination of above features, for instance adversary could be globally passive, while locally active.
In a standard network model all nodes are homogenous in the sense that all have the same characteristics (e.g, [16]). We list two main assumptions on the network, which should make this point more clear:

a. traffic flows may be the same in all nodes. They can receive, send or forward traffic;
b. there is constant capacity of every link in the network. In other words the number of messages that can pass over any link is the same.

We do not make any further assumption on the network topology. We even do not assume existence of mixes. The reason for such high level of abstraction is the need to be able to handle traffic analysis for wireless networks, [18]. It is also general enough for other types of abstract networks and paradigms, for instance: packet networks, ad-hoc networks, real-time systems and mobile agent systems.

3.2 Attacks

A system is only as secure as its weakest link. (common knowledge)

We present a few generic classes of attacks, which are not based on particular implementations.
Such an approach implies no implementation weakness, which could be opportunistically exploited. As the result all cryptographic protocols involved should work perfectly, which in real life is a very rare case; see [19]. However, some of the attacks might require additional assumption about the network. For instance, if somehow you try to abuse the mix, obviously you need it in the network, which you attack.
We rather omit simple traffic analysis attacks (although a brute force attack might fall into this category), assuming that ideas behind them were already outlined in the introduction. We focus on more sophisticated attacks on the network, which has at least some basic level of protection.
There are many ways to classify the attack types, in this paper we follow one described in [9]. The attacks are:

a. *brute force attack.* The idea is to follow every possible path for the message. When attacker is lucky enough she can link sender and recipient, in other case she can at least narrow list of possible recipients. All that an attacker needs to do is taping on the required wires, so his sufficient characteristics is: static, local, passive, external. Such a method might be a great strain on attacker's resources, but if you assume computationally unbounded adversary or well funded government agency it still might be feasible;
b. *Flooding/Node Flushing/Spam attack.* The idea of this attack is attributed to Chaum. The node is flooded with a number of messages sufficient to make it forward the messages (e.g. by exhausting node's storage). Properly performed this type of attack allows to associate messages leaving the node with ones that entered. Sufficient characteristics for the adversary is: static, local, active, external;
c. *Timing attack.* The attack is based on the observation that different routes in the network might require different times to travel. This property might allow to differentiate between various messages. Sufficient characteristics for the adversary is: static, passive;
d. *Contextual attacks*. The whole class of attacks, which prey on the fact that real user's behavior is difficult to model precisely. Members of the class are:
    - communication pattern attacks;
    - packet counting attacks;
    - insertion attacks.

   This class of attacks is especially effective, when used against real-time networks;
e. *Denial of Service attack*. By making some nodes unavailable adversary might make easier obtaining information on messages' routes, especially if the attack is combined with some other attack methods;
f. *Active attacks exploiting user reactions.* User behavior might depend on message received, the clue is to find the behavioral pattern;
g. *Message Tagging*. Messages are modified, so the adversary controlling some points in the network (e.g., the first and the last node) can spot them.

Although we tried to keep our model abstract, please note that still some of the attacks are not applicable for all abstract network types, e.g. wireless environment.
From the brief description provided above it should be pretty much clear that simple protection mechanism do not solve all the problems. Now, it is the time to have more detailed look at countermeasures.

4. Countermeasures

Traffic analysis attack is very difficult to note. This is especially true for passive observation attack, since it leaves no trace. Hence, it is a prudent practice to assume that traffic is under a constant observation and act accordingly.

A majority of the schemes' first and often only line of defense is anonymity (e.g. [20]). Although there is a whole continuum for degrees of anonymity (see [21], [14]), to simplify the model it is assumed that it is a binary value and that it can be lost only once. While in some paradigms (e.g., mobile agents, see [22]) its is still possible to protect principals in case of anonymity loss, in paper we focus on the first line of defense.

4.1 Basic methods

First way to ensure privacy is obviously encryption (e.g. [1]). However, it protects only privacy of the message payload, but not the traffic pattern.

The first idea to counter traffic analysis was to use *padding*. At first padding concerned the message length, since having all encrypted messages of the fixed length would be a considerable advantage. However, communication protocols (e.g., TCP/IP) have much better performance using messages of different sizes. Hence, we witness usual tradeoff in cryptography: security vs. performance.

Padding can be also use in the traffic context. The idea behind *traffic padding* is to make number of true traffic patterns very large, so any payload traffic flow can correspond to a predefined pattern, see [23]. The padding method has two variants: end-to-end flow cover or link cover (e.g., [24]). While simple in theory, in practice traffic padding suffers from uncontrollable disturbances in the system, which may leak information. This problem can be managed to some extend by statistical pattern recognition (see [23]), which should provide information on the disturbance significance.

Increasing the total amount of the traffic is one of the methods to randomize communication patterns. This is often achieved by introducing dummy traffic, see [25]. One of practical solutions in this field is VAST ([26]) , which provides anonymity in WWW system.

Padding can be also used to prevent some of active attacks. For instance, consider ping probing (see [17]); the variant of a timing attack, which can be blocked by randomly delaying the non-payload traffic (like *ping*).

However, padding has several drawback. First, it requires certain nodes (actually in some implementations all nodes) to be reliable and secure. Secondly, it consumes the bandwidth and requires to saturate communication links even, when there is no traffic to be send. This can be a major drawback in Ad Hoc (see [27]) and/or radio networks (see [24]), where bandwidth and/or power supply might be limited. Things get worse, if one takes into account that in such a networks nodes have to broadcast (without acknowledgement) the message to all other nodes in range (see [18], [24]) in order to hide message headers.

Third issue, often acknowledged by the method designers (e.g., [26]), is that an untypical traffic patterns still can be extracted from the ocean of data flow, see [7].

Routing is second major method, which is used to protect against traffic analysis. The idea is to make messages (packets) traveling by the different ways, with anonymizing services available at least at some nodes of the network. Such an approach was for the first time proposed by Chaum in the context of anonymous e-mail service ([12]). The key element was a Chaum's mix, a node, which for all incoming messages randomizes (mixes) them on the output. Similar idea is behind Chaum dinning cryptographers, see [13]. Other interesting approach is Crowds ([14]), where users are blended (mixed) in big and diverse groups to anonymously exchange messages. Solutions from this category often employ cryptography, with perfect example being onion routing, see [15]. The name comes from the layers of encryption that are stripped off the message by consecutive nodes. As it was nicely written by Paul Syverson: "Onion routing operates by dynamically building anonymous circuits within a network of onion routers, similar to real-time Chaum mixes." ([28]).

While routing is a good solution it suffers from certain drawbacks. First, the traffic has to take place in the large enough network (number of nodes and connections between them), so the number of possible combinations is sufficiently big. Secondly, taking different routes will result in a different times of arrival. This can be window of opportunity for an attacker (e.g. timing attacks), but also a major design problem in real-time networks, see [29].

Other problems might appear in Ad Hoc networks, where mix nodes might be unreachable and bandwidth cannot be guaranteed, see [27].

4.2 TA metrics

In the first approach to traffic analysis the focus was on how well actions of a particular user were protected. Traffic analysis prevention (TAP) metrics ([16]) answers a little different question. It provides information on how much protection is available to all the users of the system collectively. To quote the original paper ([16]): "previous work has focused on how well the system distributes available anonymity, while we focus on the amount of anonymity there is to distribute.".

In order to describe the concept one has to start with traffic matrix (TM). TM is a representation of all end-to-end flows in the network.

The goal of TAP is to hide actual TM from the adversary. It is achieved by making the set of TM resulting from adversary observations large enough. The adversary does not have any a priori means of excluding any of these TMs, hence probability of finding the right TM in the set can be made sufficiently low. It is an entropy based approach to the amount of uncertainty that adversary faces while trying to determine actual TM.

The concept allows to bring into the picture various padding and routing strategies and treat them as matrix operations. It also allows to associate costs to the flow and evaluate various strategies according to imposed cost constrains.

4.3 TA tradeoffs

Previously during discussion on various aspects of traffic analysis we provided hints on particular solutions' limitations and tradeoffs involved. Now, it is the time to put all this information together.
While many metrics for the tradeoffs can be designed (e.g. [16]), following [5] we tend to view traffic analysis as an optimization problem. The optimization criteria would be the following:

- traffic analysis resistance;
- performance;
- resistance to catastrophic DoS or DDoS;
- bandwidth cost;
- resources (power supply).

An optimization problem has to be solved given particular network constrains. Although, we tried to be as general as possible, it cannot be excluded that particular network type will bring some additional factors into the picture. For instance, let's consider packet radio networks, which we mentioned briefly in Section 4.1, while talking about padding. Preventing traffic analysis requires to broadcast without acknowledgement the message to all other nodes in range (see [18], [24]). But this can result in poor reliability of the network and possibly deteriorate end-to-end performance.
While the solution to this problem would be to introduce a protocol with multiple broadcast, this would result in possible network slowdown and increased power consumption. In turn this would be a major problem in mobile applications, where power management is crucial.
Another example would be real-time networks, where real-time requirement has got priority over all other optimization criteria and constrains, see [29].
Last but not least, use of TA countermeasures can result in creating new side channels. A side channel attack (a.k.a covert channels interface) uses some secondary data about the object investigated to deduce its main properties. In a case of mobile agents' traffic analysis such a phenomenon was described as an Overprotection Paradox and is a serious threat, see [22].

5. Conclusions

In the paper we studied problem of preventing traffic analysis. We started from real-life origins of TA, since as a time goes by, cyberspace and reality are converging. Such an approach allowed of present the main idea behind TA attacks.
Network and adversary assumptions, together with descriptions of some attacks, allowed us to discuss countermeasures against traffic analysis. We pointed to the limitations of the particular methods and discussed new approach based on traffic analysis prevention (TAP) metrics. This approach seems to be a new field of research with interesting open problems.
TA prevention is a complex task with many limitations, which need to be taken into account. Some of these result from the network type, while the other fall in one of general tradeoffs' category.
To emphase this point it seems instructive to bring the following quote from Schneier: "... thinking of security not in absolutes, but in terms of trade-offs -- the inevitable expenses, inconveniences, and diminished freedoms we accept (or have forced on us) in the name of enhanced security. Only after we accept the inevitability of trade-offs and learn to negotiate accordingly will we have a truly realistic sense of how to deal with risks and threats." ([30]).
In this context it is interesting to recall an Overprotection Paradox: an excessive use of countermeasures against traffic analysis produce side channels, which will benefit the attacker.
A proper handling of this paradox is one of open research problems. It is not sure whether it can be solved once and for all, because it looks very much like an example of an old sword versus shield competition. Interplay between reality and cyberspace, which introduces new dimensions and side channels in TA (e.g., mobile agents), makes it more difficult and challenging.

References

[1] Bruce Schneier. "Applied Cryptography". John Wiley & Sons, New York. 1996.

[2] Peter Wright., "Spycatcher: The Candid Autobiography of a Senior Intelligence Officer", Viking, New York, 1987.

[3] Ross J. Anderson. "Security Engineering: A Guide to Building Dependable Distributed Systems". John Wiley & Sons, New York. 2001.

[4] "Wall Street", the movie, directed by Oliver Stone, Twentieth Century Fox, 1987.

[5] Adam Back, Ulf Moller, and Anton Stiglic. Traffic Analysis Attacks and Trade-Offs in Anonymity Providing Systems. In: I. S. Moskowitz (Ed.): IH 2001, LNCS 2137, Springer-Verlag Berlin Heidelberg , pp. 245–257, 2001.

[6] http://www.safeweb.com

[7] Andrew Hintz. Fingerprinting Websites Using Traffic Analysis. In: R. Dingledine and P. Syverson (Ed.), Privacy Enhancing Technologies 2002, pages 171–178. Springer-Verlag, LNCS 2482, 2003.

[8] Andreas Pfitzmann and Marit Kohntopp. Anonymity, unobservability and pseudonymity — a proposal for terminology. In: Hannes Federrath (Ed.), Designing Privacy Enhancing Technologies: Design Issues in Anonymity and Observability, pages 1–9. Springer-Verlag, LNCS 2009, 2000.

[9] Jean-Francois Raymond. Traffic Analysis: Protocols, Attacks, Design Issues, and Open Problems. In: H. Federrath (Ed.): Anonymity 2000, LNCS 2009, Springer-Verlag Berlin Heidelberg, pp. 10-29, 2001.

[10] R. Cramer, I. Damgard. Multiparty Computations, An Introduction. In: Advanced Course on Contemporary Cryptology, Centre de Reserca Matematica, Barcelona 2004.

[11] Benny Chor, Oded Goldreich, Eyal Kushilevitz, and Madhu Sudan. Private information retrieval. Journal of the ACM, 45(6) pp. 965-981, 1998.

[12] David Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. Communications of the Association for Computing Machinery 24 (2), pp. 84-88, 1981.

[13] David Chaum. The Dining Cryptographers Problem: Unconditional sender and recipient untraceability. Journal of Cryptology 7 (1), pp. 65-75, 1988.

[14] M.K. Reiter, A.D. Rubin. Anonymous Web transactions with crowds. Communications of the ACM 42 (2), pp. 32-48. 1999.

[15] P. Syverson, G. Tsudik, M. Reed, C. Landwehr. Towards an analysis of onion routing security. In: Proc. Workshop on Design Issues in Anonymity and Unobservability, ICSI RR-00-011, pp. 83-100. 2000.

[16] Richard E. Newman, Ira S. Moskowitz, Paul Syverson, Andrei Serjantov. Metrics for Traffic Analysis Prevention. In: R. Dingledine (Ed.): PET 2003, LNCS 2760, Springer-Verlag Berlin Heidelberg, pp. 48–65, 2003.

[17] Xinwen Fu, Bryan Graham, Riccardo Bettati and Wei Zhao. Active Traffic Analysis Attacks and Countermeasures. In: Proceedings of the 2003 International Conference on Computer Networks and Mobile Computing (ICCNMC'03) IEEE.2003.

[18] M. Blaze, J. Ioannidis, A. Keromytis, T. Malkin, and A. Rubin. Protocols for anonymity in wireless networks. In: B. Christianson at al., eds., Post-proceedings of the 11th Cambridge International Workshop on Security Protocols, Sidney Sussex College, 2-4.04.2003, LNCS, Springer-Verlag, Berlin, 2004 (in print).

[19] Ross Anderson, Roger Needham. Programming Satan's computer. In: Computer Science Today, LNCS vol. 1000, Springer-Verlag. 1995.

[20] Beimel, A., and S. Dolev. Buses for anonymous message delivery. Journal of Cryptology 16, pp. 25–39, 2003.

[21] Claudia Diaz, Stefaan Seys, Joris Claessens, and Bart Preneel. Towards measuring anonymity. In: Paul Syverson and Roger Dingledine, editors, Privacy Enhancing Technologies (PET 2002). Springer-Verlag, LNCS 2482, April 2002.

[22] K.Kulesza, Z.Kotulski, K.Kulesza. On mobile agents resistant to traffic analysis. In: Maurice H. ter Beek, Fabio Gadducci, [eds.], Proceedings of VODCA 2004, First International Workshop on Views On Designing Complex Architectures, Bertinoro, Italy, 11-12 September 2004.

[23] Xinwen Fu, Bryan Graham, Riccardo Bettati and Wei Zhao. On Countermeasures to Traffic Analysis Attacks. In: Proceedings of the 2003 IEEE Workshop on Information Assurance, United States Military Academy, West Point, NY.2003.

[24] Shu Jiang, Nitin H. Vaidya, Wei Zhao. Preventing Traffic Analysis in Packet Radio Networks. DARPA Information Survivability Conference and Exposition (DISCEX II'01).

[25] O. Berthold and H. Langos. Dummy traffic against long term intersection attacks. In Paul Syverson and Roger Dingledine, editors, Privacy Enhancing Technologies (PET 2002). Springer-Verlag, LNCS 2482, April 2002.

[26] Igor Margasinski, Krzysztof Szczypiorski - VAST: Versatile Anonymous System for Web Users. In: Proc. of The Tenth International Multi-Conference on Advanced Computer Systems ACS'2003, October 22-24, 2003 Miedzyzdroje, Poland.

[27] Sungchang Lee, Ha Young Yun, Mi Lu. Load Balanced Onion Relay for Prevention of Traffic Analysis in Ad Hoc Networks. In: H.-K. Kahng and S. Goto (Eds.): ICOIN 2004, LNCS 3090, pp. 24–33, 2004.

[28] Paul Syverson. Onion Routing for Resistance to Traffic Analysis. In: Proceedings of the DARPA Information Survivability Conference and Exposition (DISCEX'03), IEEE. 2003.

[29] Yong Guan, Chengzhi Li, Dong Xuan, Riccardo Bettati, Wei Zhao. Preventing Traffic Analysis for Real-Time Communication Networks. Proceedings of the IEEE MILCOM, Atlantic City, NY. 1999.

[30] Bruce Schneier. Crypto-Gram Newsletter August 15, 2003. Quote is taken from Schneier's remarks on his new book "Beyond Fear". http://www.schneier.com/crypto-gram-0308.html.