Steganographic Methods of Communications in Distributed Computing Networks

Artem S. Konoplev
Information Security Center, Peter the Great
St.Petersburg Polytechnic University,
St. Petersburg, Russia
+7 812 552 76 32
artem.konoplev@ibks.ftk.spbstu.ru

Alexey G. Busygin
Information Security Center, Peter the Great
St.Petersburg Polytechnic University,
St. Petersburg, Russia
+7 812 552 76 32
info@ibks.ftk.spbstu.ru

ABSTRACT

This paper reviews the problem of a secure data transfer in distributed computing networks. It analysis the most popular covert channels (the steganographic methods of communications) and introduces their classification. The article also presents a class of the most effective steganographic methods, describes its formal model and performs a security analysis based on the proposed model.

Categories and Subject Descriptors

C.2.0 **[Computer-Communication Networks]**: General – *Security and Protection*

General Terms

Algorithms, Security

Keywords

Distributed computing networks, the Grid, information security, steganography, covert channel, network protocols.

1. INTRODUCTION

The intensive growth of data processing in the Internet leads to the wide integration of the high-performance intelligent systems allowing storing, transmitting and analyzing a big variety of heterogeneous information such as user content, control messages, audio and video flows, multimedia content, etc. Such systems are based on distributed computing networks (for example the Grid) and used for the solutions in the scientific and economic spheres. A data transfer process in the distributed computing networks has the following features:

- takes place in a heterogeneous environment (both wired and wireless);
- is performed by a complex network protocol stack;
- is supervised by a complex network security and management systems (DPI, DLP, SIEM etc.) [14, 17].

The specified features determine a need for the steganographic methods of communications (SMC). These methods provide the confidentiality of a transmitted data and have an advantage over cryptographic techniques [16] because the use of encryption cannot hide the fact of the existence of a data transfer process and therefore in some cases allows an attacker:

- to identify the source and destination of a transmitted messages;
- to make an assumption about the occurrence of some significant event (this assumption could be based on the traffic parameters change tracking).

Accordingly it is important to perform the effectiveness analysis of the modern SMC using the following criteria:

- bandwidth;
- implementation and maintenance complexity;
- detection complexity.

In this article we have performed an analysis of the well-known SMC and determined the most effective one.

2. RELATED WORK

One of the most popular implementations of the communication channels using the SMC is the case when the covert sender generates its own network traffic and masquerades it as one of the widely used network protocols. Such SMC are referenced as active [1].

The straightforward SMC implementation approach is an encapsulation of the covert data into the headers of a network protocols' packets. This approach could be applied on any level of the OSI/ISO model.

Covert data could be encoded into the following fields of the TCP/IP stack packets [1, 3]:

- unused or reserved bits;
- source/destination address;
- source/destination port;
- packet length;
- IP and TCP timestamps;
- IP packet ID and Offset fields;
- IP packet TTL field;
- checksums;
- TCP initial sequence number;
- padding.

Covert data could also be transmitted inside an upper layer protocols such as HTTP [4]. The examples of some simple encoding techniques are given in table 1.

Table 1. Encoding scheme based on direct data insertion

HTTP Header	Encoded Value
cookie: sid=covertdata	covertdata
customheader: covertdata	covertdata

The examples of SMC utilizing other application layer protocols can also be found in [5] for DNS and [6] for Skype.

S. Cabuk examined a SMC based on timing intervals [7]. In proposed method a timeline is divided on equal timing intervals. Sending packet during the timing interval is

SIN '15, September 08 - 10, 2015, Sochi, Russian Federation
© 2015 ACM. ISBN 978-1-4503-3453-2/15/09…$15.00
DOI: http://dx.doi.org/10.1145/2799979.2800024

equivalent to sending "1". Accordingly, not sending packet during the timing interval is equivalent to sending "0".

As opposed to active SMC there are passive ones [1].

A covert channel which implements the passive SMC operates between two gateway devices such as routers, bridges, firewalls etc. These devices don't generate their own network traffic but encode a covert data into a transit traffic generated by their network environment.

J. Rutkowska proposed the Linux kernel based implementation of a passive covert channel using the NUSHU protocol [2]. This protocol specifies a covert data encoding technique whereby covert data is written into a TCP initial sequence numbers.

It should be mentioned that most of the SMC are medium independent.

3. SMC CLASSIFICATION

We introduced a SMC classification based on S. Zander covet channel taxonomy [1]. Fig. 1 illustrates our classification criteria.

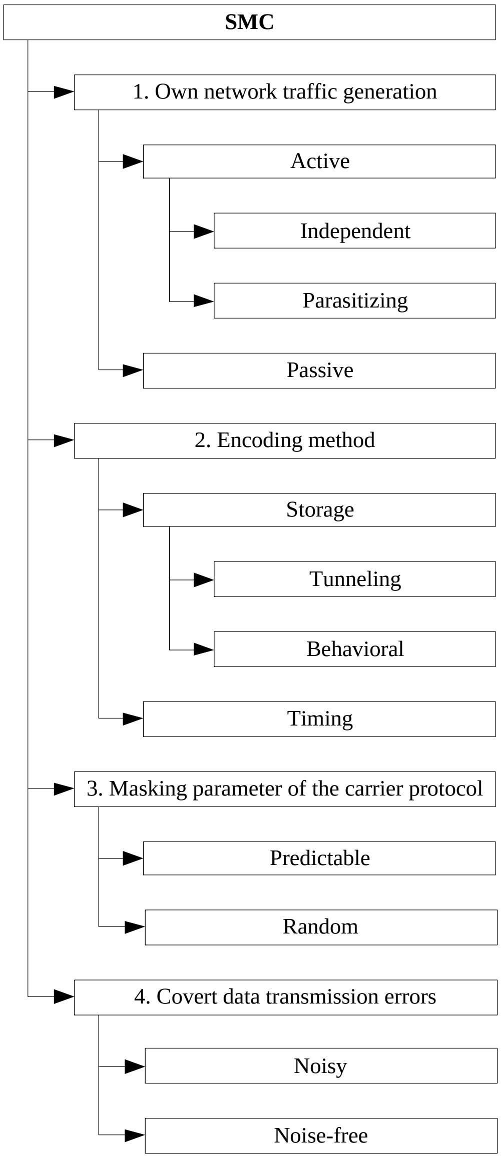

Figure 1. SMC classification criteria.

Covert sender utilizing *active* SMC generates its own traffic used for covert data encoding. Active SMC are classified into independent and parasitizing. *Independent* SMC sender is responsible for carrier protocol operation maintenance (packet building, session control etc.). *Parasitizing* SMC sender utilizes third-party software that is used by other "normal" system users. In this class of SMC control over the carrier traffic is sacrificed for covertness. A covert sender implementing the *passive* SMC acts as a middleman. It does not generate its own traffic but utilizes a transit traffic in purpose of covert data encoding.

Timing SMC's encode covert data using the timing intervals between the carrier protocol events. *Storage* SMC's encode covert data using a content of the carrier protocol messages (packets, signals etc.). Covert data could be explicitly inserted into the carrier protocol messages. In this case storage SMC's are called *tunneling*. On the other hand covert data could be encoded implicitly as a sequence of the carrier protocol events. Such SMC's are called *behavioral*.

All SMC's encode covert data by changing several parameters of the carrier protocol (cover). SMC is called *random* if cover is a pseudorandom variable. Otherwise SMC is called *predictable*.

In some cases carrier protocol could not provide error correction of the transmitted covert data. Hence SMC have to ensure the covert data integrity and perform error corrections by itself. Such SMC's are called *noisy*. The SMC's that rely on carrier protocol error corrections are called *noise-free*.

4. SMC EFFECTIVNESS ANALYSIS

We performed the assessment of the well-known SMC implementations' bandwidth. We declared the carrier protocol operation rate equal 1 Mbps (250 Kbps for ZigBee networks) and the average packet length equal 500 bytes. Also we estimated the SMC implementation complexity by the following scale:

- low – the carrier protocol specification and SMC implementation tools are publicly available; covert sender and receiver have no restrictions on their operation (for example relative location restriction);
- medium – the carrier protocol specification and tools for SMC implementation are publicly available but there are restrictions on covert sender or receiver operation;
- high – the carrier protocol specification or tools for the particular SMC implementation are publicly unavailable.

The assessment results are presented in the table 2.

Table 2. SMC bandwidth and implementation complexity assessment

SMC Implementation	Class	Bandwidth, Kbps	Comple-xity
NUSHU [2]	Passive, tunneling, random, noise-free	0.24	Medium
IP and TCP header fields [1, 3]	Active, tunneling, predictable, noise-free	29.5	Low
HTTP headers [4]	Active, tunneling, predictable, noise-free	> 64	Low
Infranet [13]	Active, behavioral, random, noise-free	> 64	Low

SMC Implementation	Class	Bandwidth, Kbps	Comple-xity
DNS TTL [5]	Active, tunneling, random, noise-free	18	Low
SkypeMorph [6]	Active, tunneling, random, noise-free	> 64	High
RSTEG [8]	Active, tunneling, predictable, noisy	50	Low
WiFi rate switching [9]	Active, timing, random, noise-free	0.06	Medium
WiFi jammer [10]	Active, timing, random, noisy	0.0004	Medium
WiFi WEP IV [11]	Active, tunneling, random, noise-free	6	Low
ZigBee frame header fields [12]	Active, tunneling, random, noise-free	5,56	Low

The performed analysis shows that that the SMC's utilizing wireless medium features are less effective than the upper layer methods. Also it allows us to distinguish the class of the most effective SMC's: active behavioral random noise-free methods.

5. SMC DETECTION COMPLEXITY ANALYSIS

In the previous section we differentiated the class of the most effective SMC's. We examined detection complexity of this SMC class and introduced its formal model.

Let the $X = \{x_1, x_2, \dots, x_k\}$ be the original alphabet of cardinality $\#X = k \geq 2$ which is used to represent covert data. All possible words with length of n characters from X belong to the set X^n.

Let the $Y = \{y_1, y_2, \dots, y_l\}$ be another alphabet of cardinality $\#Y = l \geq k^n$ which is used to represent the carrier protocol events.Y can be divided on $k^n + 1$ subsets $\{Y_1, Y_2, \dots, Y_{k^n+1}\}$ satisfying the following requirements:

- $\#Y_i \geq 1$ for all $1 \leq i \leq k^n$;
- $\#Y_{k^n+1} \geq 0$;
- $Y_i \cap Y_j = \emptyset$for all $1 \leq i \leq (k^n + 1)$,$1 \leq j \leq (k^n + 1)$, and $i \neq j$.

In case $l > k^n$ an example of such a division can serve the next one:

$$\{Y_1 = \{y_1\}, Y_2 = \{y_2\}, \dots, Y_{k^n} = \{y_{k^n}\}, Y_{k^n+1} = \{y_{k^n+1}, y_{k^n+2}, \dots, y_l\}\}.$$

Obviously, such a division is not unique. In case $l = k^n$ an example of such a division can serve the following:

$$\{Y_1 = \{y_1\}, Y_2 = \{y_2\}, \dots, Y_{k^n} = \{y_{k^n}\}, Y_{k^n+1} = \emptyset\}.$$

Hence the following bijective mapping can be set:

$$\varphi: X^n \to \{Y_1, Y_2, \dots, Y_{k^n}\}.$$

Thus every word from X^n is uniquely associated with a subset from Y. There is one-to-one relation. So every covert data word can be encoded with one of the carrier protocol events.

If $Y_{k^n+1} \neq \emptyset$ then events from Y_{k^n+1} can be used as idle symbols in purpose of increasing the SMC detection complexity. These symbols will be ignored in decoding process.

Let an original covert message is composed of characters from alphabet X. The message length is m which is a multiple of n (if it is not message padding is applied). Before sending the original message has to be encoded with characters from the alphabet Y. The encoding is performed using the following algorithm.

Input: message $d \in X^r$, $r \geq n$, $r \equiv 0 (\mathrm{mod}\, n)$, and bijective mapping $\varphi: X^n \to \{Y_1, Y_2, \dots, Y_{k^n}\}$.

Output: encoded message $s \in Y^{r/n}$.

1. Divide message d on r/n words from X^n:

$$d = (d_1, d_2, \dots, d_{r/n}), d_i \in X^n, 1 \leq i \leq \frac{r}{n}.$$

2. Compute$s' \in \{Y_1, Y_2, \dots, Y_{k^n}\}^{r/n}$:

$$s' = \left(\varphi(d_1), \varphi(d_2), \dots, \varphi(d_{r/n})\right).$$

3. For each $\varphi(d_i) \in \{Y_1, Y_2, \dots, Y_{k^n}\}$, $1 \leq i \leq \frac{r}{n}$, select arbitrary character $s_i \in \varphi(d_i)$. Encoded message is:

$$s = (s_1, s_2, \dots, s_{r/n}).$$

Original message d can be restored from encoded message s with the inverse mapping.

The given algorithm can be extended with insertions of idle characters from subset Y_{k^n+1}.

Let us consider the special case of the introduced model where

$$\varphi: X^n \to Y$$

(monoalphabetic substitution case).

Let us consider the timing interval length t. During this interval carrier protocol operation is performed and u characters from Y are sent (u carrier protocol events occurre). In this case covert sender performs covert message transmission with average rate $\omega_Y = \frac{u}{t}$ characters from Y per time unit or $\omega_X = \omega_Y n$ characters from X per time unit.

Let $\overline{\omega_X}$ be a minimal acceptable covert channel bandwidth. Hence from the inequality

$$\omega_X \geq \overline{\omega_X}$$

we can deduce that the covert channel rate is enough for covert data transmission.

Let us rewrite the inequality:

$$\omega_Y n \geq \overline{\omega_X},$$

$$n \geq \frac{\overline{\omega_X}}{\omega_Y}.$$

Therefore with fixed $\overline{\omega_X}$ every character from Y has to contain not less than $\frac{\overline{\omega_X}}{\omega_Y}$ characters of the original message.

Every character from Y (an event of the carrier protocol) is represented with a word $(z_1, z_2, \dots, z_L)$, $z_i \in Z$, from some protocol-specific alphabet Z of cardinality $\#Z$. Accordingly, each character z_i contains not less thanlog$_k$ $\#Z$ characters of the original message on the average. Therefore the average length of the word $(z_1, z_2, \dots, z_L)$ can be assessed with the following inequality:

$$L \geq \frac{n}{\log_k \#Z} \geq \frac{\overline{\omega_X}}{\omega_Y \log_k \#Z},$$

i.e.

$$L \geq \frac{\overline{\omega_X}}{\omega_Y \log_k \#Z}.$$

Let also declare the maximum allowed carrier protocol event frequency $\overline{\omega_Y}$ which can be evaluated practically by analyzing the carrier protocol during its normal operation. Similarly, let us declare $\overline{L}$ as the maximum allowed L value.

The following inequality system allows us to distinguish the carrier protocol event sequences which can be used for confidential covert data transmission.

$$\begin{cases} L \geq \dfrac{\overline{\omega_X}}{\omega_Y \log_k \#Z} \\ L < \overline{L} \\ \omega_Y < \overline{\omega_Y} \end{cases}.$$

Fig. 2 is a graphical representation of the introduced inequality.

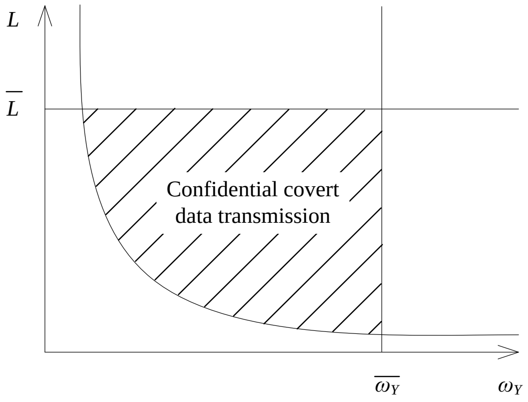

Figure 2. Events that can be used for secure covert data transmission.

The term “confidential” in current context means that the active behavioral random noise-free SMC implementation with characteristics L and ω_Y from the area represented at Fig. 4 will be hardly detectable. The detection of such a SMC could be performed using complicated methods [15] requiring from a supervisor an extra knowledge about:

- medium characteristics;
- carrier protocol utilized;
- covert data characteristics.

6. CONCLUSION

The results of our work show that application of the active behavioral random noise-free SMC’s allows to provide the high level of confidentiality and acceptable bandwidth for communications in the distributed computing networks.

7. ACKNOWLEDGMENT

Project is financially supported by Ministry of Education and Science of Russian Federation, state assignment for educational institutions of higher education within the jurisdiction of the Ministry of Education of Russia, in scientific area (contract No.2.1778.2014/K; July 15, 2014; application No.1778).

8. REFERENCES

[1] Zander, S. 2010. *Performance of Selected Noisy Covert Channels and Their Countermeasures in IP networks.* Ph.D. dissertation, Centre for Advanced Internet Architectures, Swinburne University of Technology.

[2] Rutkowska, J. 2004. The implementation of Passive Covert Channels in Linux Kernel.In *21st Chaos Communication Congress* (Berlin, Germany, December 2004).

[3] Yuan, B., and Lutz P. 2005. A Covert Channel in Packet Switching Data Networks. In *Proceedings of the Second Upstate New York Workshop on Communications and Networking* (Rochester, New York, USA, 2005).

[4] Smeets, M., and Koot, M. 2006. *Research Report: Covert Channels.* Master’s thesis, University of Amsterdam.

[5] Hoffman, C, Johnson, D., Yuan, B., and Lutz P. 2012. A Covert Channel in TTL Field of DNS Packets. In *The 2012 International Conference on Security and Management* (Las Vegas, USA, July 16-19, 2012).

[6] Moghaddam, H. 2013. *SkypeMorph: Protocol Obfuscation for Censorship Resistance.* Master’s thesis, University of Waterloo.

[7] Cabuk, S., Brodley C., and Shields C. 2004. IP Covert Timing Channels: Design and Detection. In *11th ACM Conference on Computer and Communications Security* (New York, USA, 2004).

[8] Mazurczyk, W., Smolarczyk, M., and Szczypiorski, K. 2010. Retransmission Steganography and Its Detection. In *Soft Computing, Journal no. 500 Springer.*

[9] Calhoun, T., Cao, X., Li, Y., and Beyah R. 2012. An 802.11 MAC Layer Covert Channel. *In Wireless Communications and Mobile Computing.* Volume 12, Issue 5, 393-4005 (April 2012).

[10] Shah, G., and Blaze, M. 2009. Covert Channels through External Interface. In *Proceedings of the 3rd USENIX Conference on Offensive Technologies* (Montreal, Canada, July 31-August 04, 2009).

[11] Blanco, A., and Gutesman E. 2011. Abusing the Windows WiFi Native API to Create a Covert Channel. In *Hack.lu* (Lexenbourg, September 20, 2011).

[12] Martins, D., and Guyennet H. 2010. Attacks with Steganography in PHY and MAC Layers of 802.15.2 Protocol. In Systems and Networks Communications.

[13] Feamster, N., Balazinska, M., Harfst, G., Balakrishnan, H., and Karger D. 2002. Infranet: Circumventing Web Censorship and Surveillance.*In Proceedings of the 11th USENIX Security Symposium* (San Francisco, 2002).

[14] *Network Security Computer Appliance “RSCB-X”.* DOI= http://www.neo-bit.ru/certification.html.

[15] Markov, Y. A., Kalinin, M. O., and Zegzhda D. P. 2010. A Technique of Abnormal Behavior Detection with Genetic Sequences Alignment Algorithms. In *International Conference on Enterprise Information Systems and Web Technologies* (Orlando, Florida, USA, July 12-14, 2010).

[16] Zegzhda, D. P., Kalinin, M. O., Konoplev, A. S., and Dzyoba, A. V. 2012. The High-Performance Cryptographic Gateway for Distributed Systems of Electronic Services. In *Information Security Problems. Computer Systems.* ISSN 2071-8217.

[17] Pechenkin, A. I., and Lavrova, D. S. 2013. Parallel Network Traffic Processing on Multiprocessor Clusters for Security Analysis of Transferred Objects. In *Information Security Problems. Computer Systems.* ISSN 2071-8217.