Detection and elimination of covert communication in Transport and Internet layer – A Survey

D. M. Dakhane
Department of Computer Science & Engineering And IT,
Sipna's College of Engineering & Technology,
SGB Amravati University,
Amravti, Maharashtra, India.

Swapna Patil
Department of Electronics and Telecommunication Engineering, VIVA Institute of Technology, Mumbai University, Mumbai,
Maharashtra, India.

Mahendra Patil
Department of Computer Science & Engineering And IT,
Sipna's College of Engineering & Technology,
SGB Amravati University,
Amravti, Maharashtra, India.

ABSTRACT

Covert channels use stealth communications to compromise the security policies of systems. They constitute an important security threat since they can be used to exfiltrate confidential data from networks. TCP/IP protocols are used everyday and are subject to covert channels problems. Covert channels are used for the secret transfer of information. Encryption only protects communication from being decoded by unauthorized parties, whereas covert channels aim to hide the very existence of the communication. Initially, covert channels were identified as a security threat on monolithic systems i.e. mainframes. More recently focus has shifted towards covert channels in computer network protocols. The huge amount of data and vast number of different protocols in the Internet seems ideal as a high-bandwidth vehicle for covert communication.

The aim of this paper is to give an overview of covert channels in TCP/IP networks. We briefly describe the TCP and IP protocols, present the different types of covert channels and the methods to set them up in TCP/IP networks; then we study the existing methods to detect and eliminate covert channels.

General Terms

A **Covert Channel** may be defined as any communication channel that can be exploited by a process to transfer information in a manner that violates a system's security policy.

Steganography is the hiding of a secret message within an ordinary message and the extraction of it at its destination. Steganography takes cryptography a step farther by hiding an encrypted message so that no one suspects it exists. Ideally, anyone scanning your data will fail to know it contains encrypted data.

Keywords

Keywords— Covert channels, Steganography, TCP, IP, computer security, networking, detection, protection, analysis, traffic normalisers, packet sorting.

1. INTRODUCTION

Computer networks were basically meant for communication, connectedness and collaboration.

The notion of openness behind this revolution however, does not address the security aspect in such environments. Security issues thus finally emerged out with the pace more than the rate at which Internet has gotten in to our lives. Besides software solutions, the wedding of cryptography and network security provides concrete foundations to this active research area. Security has now become everyone's need, directly or indirectly related with network environment. This work attempts to integrate network security with another emerging technology, data hiding; primarily associated with oblivious communication or more recently protecting copyright in digital media appearing on the Internet. One of the sub-disciplines of this broad concept is covert channels which is investigated and accordingly tied with security aspects of computer networks. [3].

In this work, we attempt to identify the existence and elimination of covert communication in the TCP/IP protocol suite. We commence by giving introductory descriptions of covert channels, data hiding concepts associated with these channels and TCP/IP suite. Basic framework is then formulated.

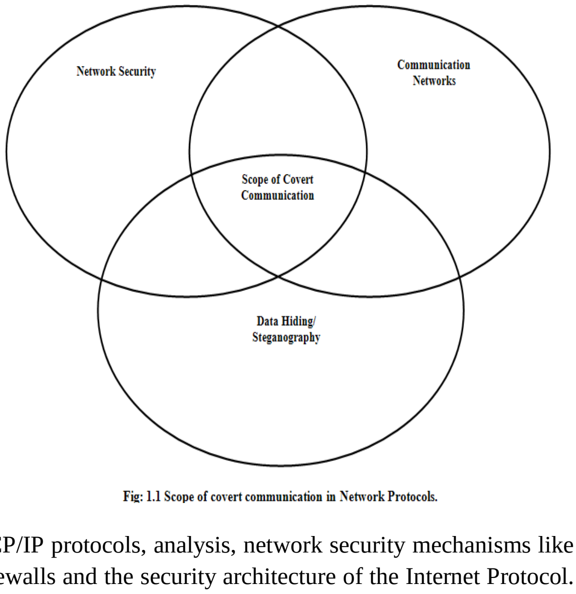


Fig: 1.1 Scope of covert communication in Network Protocols.

TCP/IP protocols, analysis, network security mechanisms like firewalls and the security architecture of the Internet Protocol. It primarily aims to provide some security means to standard network protocols and security procedures by effectively utilizing the available but *hidden bandwidth* as identified in these standard network processes.
Figure 1.1 gives a clear picture of the scope of this paper.

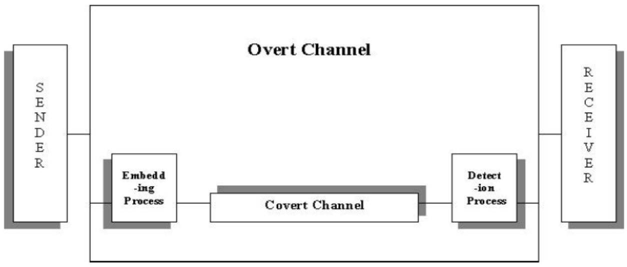


Figure 1.2 Covert Communications

Covert channels can be regarded as one of the main sub-disciplines of data hiding. In data hiding, the two communicating parties are allowed to communicate with each other based on the security policy of the system while exploiting the features as associated with covert channel definition; there is piggy-backing of undetectable data on the legitimate content. This led to an emerging discipline, *steganography*, which is the Greek for covered writing. Steganography is therefore about concealing the existence of the message when secret information is hidden into an innocent *cover* data.

This scenario would, therefore, facilitate the *smuggling* of information from one point to another. The science of steganography thus avails covert channels in order to have secret information transfer.

From a network communication point of view, these covert channels can therefore also, make use of network packets as the cover object. These network packets are shared by network nodes while traversing different network topologies before they reach their intended destination. A comprehensive approach to data hiding in the network environment should encompass network behavior as well as address data hiding aspects.

2. Literature Review & Related Work

A diverse range of individuals and groups has found reason to utilize covert channels for communication and coordination. Typically this is motivated by the existence of an adversarial relationship between two parties (such as government agencies versus criminal or terrorist organizations, hackers or corporate spies versus a company IT department, or dissenting citizens versus their governments) [1].

Clearly, government agencies, criminals, or terrorist organizations have an interest to keep their communication secret. However, simply using encryption does not prevent adversaries from detecting communication patterns. Often only the evidence that communication takes place is sufficient to detect the onset of activity, discover organizational structures or justify obtaining police warrants.

Once spies or hackers have compromised computer systems they usually ex-filtrate data or instrument the systems for malicious purposes, including communication with installed Trojan horses (malicious programs disguised as or embedded within legitimate software) or tools for launching denial of service attacks. Such activities generate network traffic that — if not covert — would immediately alert system administrators, who then would discover the compromised systems. Exfiltrating sensitive data over covert channels does not even require compromised computers. It is sufficient if the attacker can compromise an input device such as a keyboard [4], or a software package such as a web browser [5].

It should be emphasized that often even ordinary employees may want to use covert channels to bypass their company firewalls in order to access Internet resources. Furthermore, recent attempts by some governments to limit the freedom of speech in the Internet have led to proposals for using covert channels to circumvent these measures [6, 7]. In countries that forbid (strong) encryption of data, covert channels can be used to secure the information transport (although this is not strong security in the cryptographic sense).

Network administrators can use covert channels to secure network management related communication by hiding it from hackers [8]. Again this is not strong security in the cryptographic sense. Honeypots, which are computer systems set up as trap for hackers, can also use covert channels to export logged data in real-time hidden from the attacker [9].

Computer viruses or worms can use covert channels to spread themselves undetected or for covertly exchanging information necessary for distributed processing (e.g. execute brute-force attacks on cryptosystems [10]).

Covert channels can also be used for transmitting authentication data. A number of techniques have been developed for allowing authorized external users to access open firewall ports while presenting these ports as closed to all other users. One particular technique, called port knocking, uses covert channels for sending the authentication information [11]. Mazurczyk et al. proposed using covert channels and steganography to link control information, including authentication data, to the actual data flows [12, 13].

A number of researchers have developed packet traceback techniques using covert channels [14, 15]. Traceback techniques provide downstream nodes with information about the path of incoming packets. This is important in case of denial of service attacks, because it allows filtering the attack traffic at upstream nodes or even isolating the attacker(s).

2.1 Covert Channels in TCP/IP Protocol Suite:

A more specific approach is adopted by Rowland [7]. Focusing on the IP and TCP headers of TCP/IP protocol suite, Rowland devises proper encoding and decoding techniques by utilizing the IP identification field, the TCP initial sequence number and acknowledge sequence number fields. These techniques are implemented in a simple utility written for Linux systems running version 2.0 kernels. Rowland simply provides a proof-of-concept of the existence as well as the exploitation of covert channels in TCP/IP protocol suite. This work can, thus, be regarded as a practical breakthrough in this research area. The adopted encoding and decoding techniques are more pragmatic as compared to previously proposed work. These techniques are analyzed considering security mechanisms like firewall and network address translation. However, the non-detectability of these covert communication techniques is questionable. For instance, a case where sequence number field of the TCP header is manipulated, the encoding scheme is adopted such that every time the same alphabet is covertly communicated, it is encoded with the same sequence number. Moreover, the usage of sequence number field as well as the acknowledgement field can not be made specific to the ASCII coding of the English language alphabet as proposed, since both fields take in to account the sent and the receipt of data bytes pertaining to specific network packet(s).

2.1.2 Internet Steganography

Katzenbeisser and Petitcolas [16] have also observed the potential for data hiding in the TCP/IP protocol suite. The significance of using of TCP/IP stems from the sheer volume of secret communication that can be realized since TCP/IP packets are used to transport thousands of Internet packets with each overt communication link. Katzenbeisser and Petitcolas use the term Internet steganography for this potential scenario and indicate that the ongoing research work includes the embedding, recovering and detecting information in TCP/IP packet headers.

2.1.3 Covert Channel Analysis in Networks so far

The research publications discussed above:

a) Identify the existence of covert channels in a network environment.

b) Point to devising satisfactory techniques for embedding and extraction processes at the source and destination, respectively.

c) Do not consider the affect of employing covert communications on the overt communication network as a whole.

3. Analysis of Problem

A channel can be defined as a communication path by which information can flow within a computer system. A covert channel is, by contrast, a path that can allow information to flow in a manner that violates the security policy of a system, allowing the transfer of information by an unauthorised process[17].

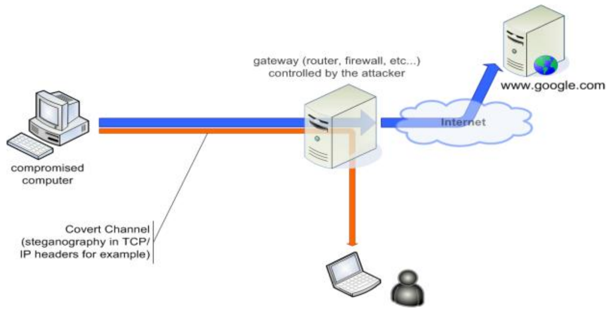

gateway (router, firewall, etc...)
controlled by the attacker
www.google.com
Internet
compromised computer
Covert Channel (steganography in TCP/IP headers for example)

Fig: Example of passive covert channel [25]

Covert channels in computer network protocols are similar to techniques for hiding information in audio, visual or textual content (steganography). While steganography requires some form of content as cover, covert channels require some network protocol as carrier.

Covert channels use stealth communications to compromise the security policies of systems. They constitute an important security threat since they can be used to exfiltrate confidential data from networks. TCP/IP protocols are used every day and are subject to covert channels problems. The aim is to give an overview of monitoring agents and elimination of covert channels in TCP/IP networks. We briefly describe the TCP and IP protocols, present the different types of covert channels and the methods to set them up in TCP/IP networks; then we study the existing methods to detect and eliminate covert channels.

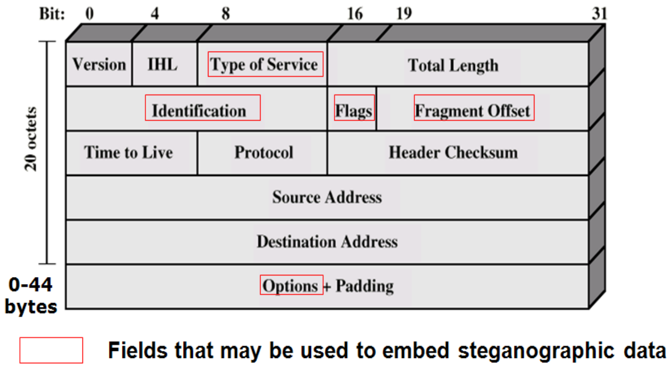
Focusing on the IP and TCP headers of TCP/IP protocol suite, Rowland devises proper encoding and decoding techniques by utilizing the *IP identification field*, the TCP *initial sequence number* and *acknowledge sequence number fields*. These techniques are implemented in a simple utility written for Linux systems running version 2.0 kernels. Rowland simply provides a proof-of-concept of the existence as well as the exploitation of covert channels in TCP/IP protocol suite[18].

■ IP allows fragmentation and reassembly of long datagrams, requiring certain extra headers

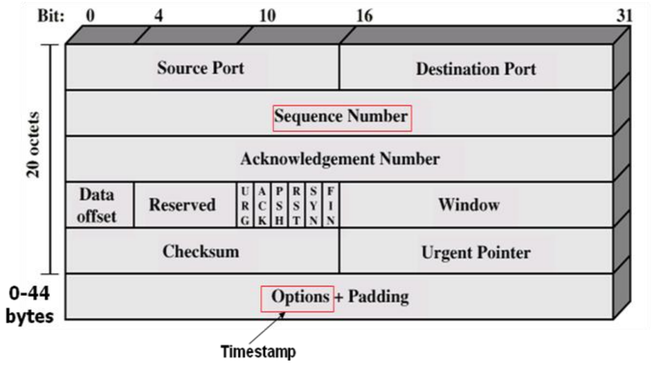
■ For IP Networks:

1. Data hidden in the IP header
2. Data hidden in ICMP Echo Request and Response Packets
3. Data tunneled through an SSH connection
4. "Port 80" Tunneling, (or DNS port 53 tunneling)

IP Header:



TCP Header:



Information is leaked by hiding data in packet header fields

- IP identification
- Offset
- Options
- TCP Checksum
- TCP Sequence Numbers

This work can, thus, be regarded as a practical breakthrough in this research area. The adopted encoding and decoding techniques are more pragmatic as compared to previously proposed work. These techniques are analyzed considering security mechanisms like firewall and network address translation.

However, the non-detectability of these covert communication techniques is questionable. For instance, a case where sequence number field of the TCP header is manipulated, the encoding scheme is adopted such that every time the same alphabet is covertly communicated, it is encoded with the same sequence number. Moreover, the usage of sequence number field as well as the acknowledgement field cannot be made specific to the ASCII coding of the English language alphabet as proposed, since both fields take into account the sent and the receipt of data bytes pertaining to specific network packet

4. Implementation

During the development of TCP/IP, little attention was paid to traditional security aspects.

For instance, the protocols governing TCP/IP are not designed to ensure integrity of the messages being transferred, nor to authenticate the originating source of the transmitted packet. A formal model of TCP/IP networks in light of some well-known security threats is presented in [19]. This model characterizes the topology of TCP/IP security to enable better understanding of the inherent vulnerabilities. Similarly, [20] points out serious security flaws in the TCP/IP protocol suite with details on a variety of attacks.

Besides identifying threats, it also presents broad-spectrum defences such as encryption. In some specific cases, introducing redundancy into the protocol specification can, in part, help protect against security vulnerabilities. In addition, there are multiple interpretations of the TCP/IP design strategies that require the natural use of redundancy.

Our work addresses the issue of the existence of covert channels within a TCP/IP environment by presenting various data hiding scenarios. Covert channels are considered as potential threats to system security. However, we consider these as unused bandwidth and accordingly suggest usage scenarios. Covert channels can be made to act as *catalyst* in various security related application usages.

In our framework, we assume that Alice and Bob, the famous analogy in cryptology, representing points A and B in an information transfer scenarios, employ data hiding involving the TCP/IP protocol suite to covertly communicate information. The covert message *Ck* traverses generally a non-ideal channel. This non-ideal channel is characterized by an *incidental process* which affects the covert message *Ck*. Keeping in view a model of the channel; Bob deciphers the covert message thereby making secret communication possible in the TCP/IP environment.

The basic framework is shown in Figure 4.1 and is explained as follows:

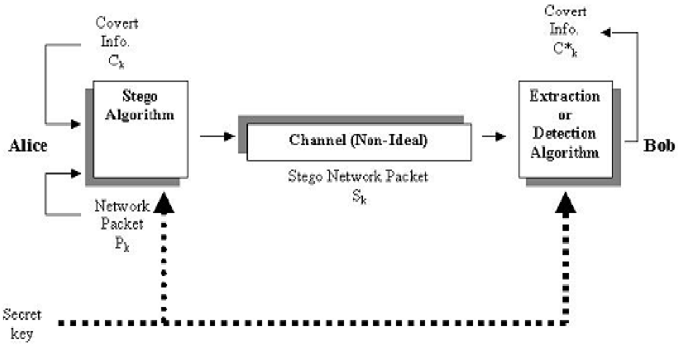


Figure 4.1 The general covert channel framework in TCP/IP

i. Cover object is the network packet *Pk*. The cover object is the data used to mask or conceal the covert information

ii. The goal of our data hiding is to produce a *stego-network packet Sk* generated by the *stego-algorithm*. Alice covertly communicates the information *Ck* to Bob by passing data through the stego network packet. She first produces this packet *Sk* (from the original packet *Pk* and *Ck*) which is transferred to Bob.

iii. There exists the possibility of a secret key known only to Alice and Bob for reasons of security.

iv. As mentioned earlier, the transmission process is modeled as a non-ideal channel representing the *secondary processing* on the stego network packet that affects the covert information flow to produce *Sk*. From the network communications perspective, this processing can introduce position error(s) in the sequence of network packets, thereby affecting the covert message, *Ck*.

v. Moreover, the same stego network packet, *Sk*, may be required to pass through an intermediate node (or multiple intermediate nodes) in order to ultimately reach Bob. As per our definition, the covert channel must not be detected by these intermediate nodes. In other words, in order to be non-detectable, any intermediate node finds no difference between *Pk* and *Sk* when processing the packet.

vi. At the intermediate node, a stego network packet, *Sk* may be dropped due to the non-availability of buffer capacity. However, this possibility is assumed to be nonexistent in our analysis of the proposed algorithms. We are focusing towards that network traffic which is most unlikely to be dropped due to buffer unavailability.

vii. Such a condition is possible as we have QoS mechanisms through which network traffic can be categorized as a preferred class. In addition, we assume there is a remote possibility that the same stego network packet is corrupted during transmission and consequently be dropped by the data link layer mechanisms.

viii. If the packet *Sk* reaches Bob, an extraction/detection algorithm is applied to the stego packet to estimate the covert information; the extracted covert information which may possibly be affected is denoted as *Ck*.

4.1 Algorithms to eliminate the covert communication in TCP/IP:

i. Eliminating covert channels in TCP/IP using active wardens [22].

ii. Eliminating TCP/IP Steganography using active warden [23].

4.2 Software tools:

A program named covert_tcp (Covert tcp was written by Craig Rowland, 1996 can be downloaded from www.packetstormsecurity.com) manipulates the TCP/IP header to transfer a file one byte at a time to a destination host. This program can act as a server and a client and can be used to conceal transmission of data inside the IP header. This is useful for bypassing firewalls from the inside, and for exporting data with innocuous looking packets that contain no data for sniffers to analyze [26].

5. Countermeasures of Covert Channels

5.1 Host Security

Securing hosts cannot remove covert network channels, but it can prevent their exploitation in some application scenarios. If hosts were secured from being hacked, the installation of Trojans, and the modification of software or the network stack would be impossible, thus hackers could not exploit covert channels. However, relying on host security could be dangerous and it would be better to eliminate covert channels in the first place where possible. Furthermore, this approach does not solve the covert channel problem in other application scenarios (e.g. censorship circumvention).

5.2 Network Security

One approach to counter covert channels is to block protocols/ports that are susceptible. Obviously, in the Internet some protocols cannot be blocked because they are vital (e.g. IP, TCP, DNS), or because their services are too important (e.g. email, Web). However, in a closed network protocols prone to covert channels could be blocked, or replaced by versions with fewer or limited covert channels.

The leakage of classified information from a high security system to a low security system (the classic covert channel) can be prevented by a network design where only hosts on the same security level are allowed to communicate. Such an approach may be practical for highly secure networks, but not for diverse large open networks such as the Internet.

Bouncing covert channels as described by Rowland [14] only work if IP address spoofing is possible. Besides solving a number of other security issues preventing IP spoofing (e.g. by ingress/egress filtering) closes such channels. Furthermore, securing networks against wiretapping, and securing routers against compromise prevents some covert channel scenarios in which covert senders or receivers act as middlemen.

5.3 Traffic Normalization

Traffic normalization can be performed by end hosts or by network devices such as firewalls or proxies.

Unused or reserved bits and padding can be dealt with easily by setting them to zero and unknown header extensions can be removed. Some covert channels exploit the fact that certain header fields are not always used (and their use is indicated by other header fields). This fact can be used for normalization as well. For example, set the IP ID to zero if the DF bit is set, set the Urgent Pointer to zero if the URG bit is not set, and set the Fragment Offset to zero if the DF bit is set. Furthermore, it should be ensured that checksums are always used and correct.

A number of other header fields can be rewritten under certain assumptions. For example, enable the DF bit and set IP ID and Fragment Offset to zero if the packet is below the MTU size (assuming the normaliser knows the MTU), rewrite the IP ID (assuming the normaliser can manipulate all fragments), rewrite the TCP ISN, source IP address and source port (assuming the normaliser can keep a mapping between original and new values and rewrite packets going in the opposite direction accordingly). Time-to-Live and TCP timestamp can also be rewritten (assuming the normaliser is located at or very close to the source) or the low order bits can be randomized.

6. Techniques to eliminate covert channels

Following are some methods to eliminate the covert channels:

6.1 Packet Regeneration

If there is a gateway, router, switch or proxy in the network which is maintaining all the protocol sessions and their states that are established by the network end systems, then there is a always a way to eliminate most of the covert channels in the protocol headers by re-packetizing the packets at these intermediate network nodes. By regenerating the received packets, we replace the traffic packet's headers with the new ones whose protocol states and most of field's values are according to the monitored session information saved at the intermediate nodes. Hence, if there are packets carrying covert message unit in some header field, then it will be replaced in the process when these packets are received by these intermediate nodes [24].

6.2 Packet Header Mangling

In this we can alter the packets in the traffic to make packets consistent in their attributes in the traffic. Hence, clearing the reserved bits, unused pointers, state bits can probably remove most covert channels. We can also prevent the covert communication by re-computing the checksums and length fields [24].

6.3 Limiting protocol support

Limiting protocol support at the switches, routers, gateways, firewalls and proxies also limit the possible covert channels and thus help in reducing covert communication [24].

6.4 Employing IDS

A network intrusion detection system is a form of passive warden that observes network traffic in search of malicious activities. An IDS with packet evasion technique exploit ambiguities in the semantics of the network protocols and the differences in perspective between IDS and end hosts. Recently, it has been proved that this kind of attacks can be defended against through the use of a protocol scrubber or a traffic normalizer which reduces ambiguous traffic to a canonical form that can be more reliably monitored [24].

7. REFERENCES

[1] Sebastian Zander and Grenville Armitage, and Philip Branch, SWINBURNE UNIVERSITY OF TECHNOLOGY MELBOURNE, AUSTRALIA "A Survey of Covert channels and countermeasures in computer network protocols", IEEE Communications Surveys & Tutorials 44 • 3rd Quarter 2007 [2] Pierre Allix, *Covert channels analysis in TCP/IP networks*, 2007

[3] "Covert Channel Analysis and Data Hiding in TCP/IP" by Kamran Ahsan

[4] G. Shah, A. Molina, and M. Blaze, “Keyboards and Covert Channels,” Proc. USENIX Security Symp., Aug. 2006.

[5] N. Vachharajani et al., “RIFLE: An Architectural Framework for User-Centric Information-Flow Security,” Proc. 37th IEEE/ACM Int’l. Symp. Microarchitecture, Dec. 2004, pp.243–54.

[6] N. Feamster et al., “Infranet: Circumventing Web Censorship and Surveillance,” Proc. 11th USENIX Security Symp., Aug.2002.

[7] C. H. Rowland, “Covert Channels in the TCP/IP Protocol Suite,” First Monday, Peer Reviewed Journal on the Internet,July 1997.

[8] D. V. Forte et al., “SecSyslog: An Approach to Secure Logging Based on Covert Channels,” Proc. First Int’l. Wksp. Systematic Approaches to Digital Forensic Engineering, Nov. 2005, pp. 248–63.

[9] The Honeynet Project, “Know Your Enemy: Sebek — A Kernel Based Data Capture Tool ,” tech. rep. , 2003,

[10] S. R. White, “Covert Distributed Processing with Computer Viruses,” Proc. 9th Annual Int’l. Cryptology Conf. Advances in Cryptology, 1989, pp. 616–19.

[11] R. deGraaf, J. Aycock, and M. Jacobson Jr., “Improved Port Knocking with Strong Authentication,” Proc. 21st Annual Computer Security Applications Conf., Dec. 2005.

[12] W. Mazurczyk and Z. Kotulski, “New Security and Control Protocol for VoIP Based on Steganography and Digital Watermarking,” tech. rep., Institute of Fundamental Technological Research, Pol ish Academy of Sciences, June 2005,

[13] W. Mazurczyk and Z. Kotulski, “New VoIP Traffic Security Scheme with Digital Watermarking,” Proc. Int’l. Conf. Computer Safety, Reliability, and Security (SafeComp), Sept. 2006, pp.170–81.

[14] E. Jones, O. Le Moigne, and J.-M. Robert, “IP Traceback Solutions Based on Time to Live Covert Channel,” Proc. 12th IEEE Int’l. Conf. Networks (ICON), Nov. 2004, pp. 451–57.

[15] H. Qu, Q. Cheng, and E. Yaprak, “Using Covert Channel to Resist DoS Attacks in WLAN,” Proc. Int’l. Conf. Wireless Networks, June 2005, pp. 38–44.

[16] S. Katzenbeisser and F. Petitcolas, Information Hiding Techniques for Steganography and Digital Watermarking. Computer Securiy Series, 685 Canton Street, Norwood, MA 02062: Artech House, Inc., 2000.

[17] "The Implementation of Passive Covert Channels in the Linux Kernel", Joanna Rutkowska,Chaos Communication Congress December 2004

[18] "Embedding Covert Channels into TCP/IP" by S.J. Murdoch, S. Lewis,University of Cambridge, United Kingdom,7th Information Hiding Workshop, June 2005

[19] G. Vigna, “A topological characterization of TCP/IP security.” Dipartmento diElettronica e Informazione, Politecnico di Milano, Piazza Leonardo da Vonci, 20133Milano, Italy, December 1996.

[20] S. M. Bellovin, “Security problems in the TCP/IP protocol suite,” Computer Communication Review, vol. 19, pp. 32–48, April 1989.

[21] T. Handel and M.Sandford., “Hiding data in the OSI network model,” (Cambridge, U.K.), First International Workshop on Information Hiding, May-June 1996.

[22] Dr. T.R. Sontakke, Sanjeev Wagh, Prashant Yawalkar, “Eliminating covert channels in TCP/IP using active wardens”

[23] Prof. D. M. Dakhane, Ms. S. R. Deshmukh, “Eliminating TCP/IP Steganography using active warden.”

[24] “Covert Channels in TCP/IP & protocol steganography”, by Kashif Ali Siddiqui 2003-03-0044.

[25] "Passive Covert Channels Implementation in Linux Kernel",by Joanna Rutkowska, Chaos Communication Congress, December 27th-29th 2004, Berlin

[26] C. H. Rowland, “Covert channels in the TCP/IP protocol suite,” *First Monday*, 1996.