Data hiding transmission using flag field in IP Header

Ahmed M.Shhab Almttaary
Department of computer Science, college of Science for women, university of Babylon

ABSTRACT.

This paper involves the manipulation of the flag field in the IP packet header for implication of a covert channel. The hidden data which are carried by the flag field are recognized at the receiver site by using a certain reference sign.

This work also attempts to overcome some difficulties that face the user of IP covert channels; these are mainly concentrated on the limitation of the message length and the recognizability of the hidden data.

The implementation has finally been done with the creation of DLL that is in charge of the lower level aspects, such as the extraction at Data Link level of the Ethernet packets like extracting the IP packets, manipulating the identification field, recalculating the checksum and resending them.

الخلاصة

يمكن استخدام قناة الإخفاء السرية (Covert Channel) لنقل المعلومات من حاسبه إلى أخرى بطريقه لا يمكن لمعترضي الرزم ملاحظتها.

تستطيع معظم انظمه الاستكشاف من الكشف بسهوله عن البيانات المخفية في الحمولة الصافية للرزمة (Payload) , ولكن يصعب عليها الكشف عن البيانات المخفية في ترويسه الحزمة (TCP/IP) .

يتضمن هذا البحث معالجة حقل العلم في ترويسه الحزمة بغيه استخدامها كقناة إخفاء سرية . ويمكن التعرف على البيانات المخفية في حقل العلم باستخدام علامة مميزة لها عند الجهة المنقولة إليها .

يتضمن البحث أيضا محاولات التغلب على بعض المشاكل التي قد تعترض مستخدمي قنوات الإخفاء السرية والتي تتركز بصورة رئيسية في تحديد طول البيانات المرسلة وطول الحزمة، وكذلك في إقرار المعلومات المخفية.

وأخيرا فقد تم في هذا البحث تنفيذ قنوات الإخفاء عبر طريق برمجة مكتبة ارتباط البيانات

المسؤولة عن دوال المستوى الأدنى مثل استخلاص حزمة IP ومعالجة حقل التشخيص وإعادة حساب فحص المجموع وإعادة الإرسال.

1.1 Introduction

Modern Computer networks, such as the internet, are designed for communication, connectedness and collaboration; their specification is “open” (i.e., publicly available) which presents difficulties with respect to security (K. Ahsan and D.Kundur, 2002). Steganography is the efficient technique to provide secure data transmission over network (R.M.Goudar,2012). In today’s computer world, it has come to mean hiding secret messages in any digital multimedia signals. Steganography works by replacing bits of useless data in regular computer files (such as graphics , sound , text , HTML) with bits of different invisible information. This hidden information can be plain text, cipher text, or even image (E.Cauich.2004). TCP/IP is the protocol used in Internet. TCP /IP were developed by a Department of Defense (DOD) research project to connect a number different networks designed by different vendors into a network of networks (the "Internet") (Miss D.D.Dhobaje , 2010). IP (Internet Protocol) is responsible for moving packet of data from node to node, and TCP (Transmission Control Protocol,) is responsible for verifying the correct delivery of data from client to server. The IP protocol defines the basic unit of data transfer through the Internet as a packet. All the data is partitioned into IP packets on the sending computer and reassembled on the

receiving computer. Each packet begins with a header containing addressing and system control information. The header packet is divided into The IP packet header consists of 20 bytes of data divided in several fields. Each field has a special purpose, depending on the type of data contained in the packet payload This approach take advantage of the unused fields of the IP header packet.. These fields are used to hide the information we want to send without raising any suspicion (Radu Ciobanu, 2011)

1.2 Covert channel

The term 'covert channel', when applied to computer networks, describes a mechanism for sending information without the knowledge of the network administrator or other users.

A communication channel is covert if it is neither designed nor intended to transfer information at all (McHugh J 2005).

A communication channel is covert (e.g., indirect) if it is based on "transmission by storage into variables that describe resource states"(M. Schaefer, B. Gold, R. Linde, and J. Scheid 2004).

Covert channels "will be defined as those channels that are a result of resource allocation policies and resource management implementation(" J. C. Huskamp 2008).

Covert channels are those that "use entities not normally viewed as data objects to transfer information from one subject to another" (R. A. Kemmerer 2007).

1.3 Related work

R.M. Goudar, Prashant N. Patil, (2012) their idea focusing on Identification field of the IP header to hide secret encrypted data, Identification field is used only when fragmentation occurs. At the receiver end, to reassemble the packets, identification field tells the right order for that. If fragmentation is not occurred, then identification field will always be unused, so that we can use this 16 bit field to hide secret encrypted message.
To avoid fragmentation, they use MTU. Maximum transfer unit decides limit for packet size for transmission over network. Sender and receiver, both should have awareness of MTU unit. For the encryption and decryption they use Elliptic curve cryptography.

Radu Ciobanu,(2011) Here the author proposes an application that reads data from a file and sends it over the covert channel which uses protocols from TCP/IP stack. The author proposes a software application that will have a loadable kernel module that checks incoming or outgoing packets for hidden data. The goal was to test several protocols that are less utilized in steganography, and to compare performances for implemented protocols. Thus SCONeP is used to send hidden data using headers from
TCP, IP, UDP and ICMP. Cryptanalysts tries to crack the encrypted data over the network, while the Steganalyst tries detecting messages that are hidden by looking at variances between bit patterns and unusually large file sizes.

Jessica Fridrich and Miroslav Goljan(2002) their idea resides in the manipulation of the IP Identification Field . The Identification Field of the IP Packet is assigned by the original sender. This number consist basically in a random number generated while the packet was being constructed. The Identification Field is only used when fragmentation occurs. Therefore; if we assure that no fragmentation will occur because of the size of the packet; it is
Possible to hide data in this field without any consequence in the transmission.

The advantages in this work are that it is used to send information from point to point, but the limitations are the quantity of information that you send. Furthermore if by any

circumstances the datagram is fragmented, the receiver will listen noise in the transmission because it will receive the same information more than one time with every new fragment of the datagram.

Chandramouli,R.; Subbalakshmi, K.P., (2003) their work is focused in the manipulation of the Do Not Fragment Bit. There is possible to indicate if we do not want that our packet be fragment by the routers in the way. In consequence; again, if we assure that our packet will be not fragmented
because of the size of it; we can hide information in the Do not fragment Bit at the flags field.
In this work the problem of the size of data is worse than the Identification Field, because here we can only transmit one bit for each datagram. Imagining that the datagram does not carry any data but the header, then the ratio useful information to total data is 1:160, it means that if you want to transmit the phrase "hello world" you will need to transmit 88 datagrams producing and overhead of almost 2 Kb for just 11 bytes.

(Buetler, 2008) A mechanism for sending and receiving information data between machines without alerting any firewalls and IDSs on the network.

1.4 Aim of work

In this paper the sender hiding the information in IP packet header using the first bit of flag that is not used (reserved). The IP packet must be controlled by hooking the Network Driver Interface Specefication (NDIS), so it can be enabled to modify the packet and therefore, to hide the information inside it. When the project execute The sender sends a key to tell the receiver that there is a hiding information and sends the length of information to the receiver and then sends the hiding information.

2. Framework

Data hiding in IP packet requires a full control of the incoming/outgoing packet. The aspect of this paper is the implementation of data hiding on the windows operating system. But the source code of the TCP/IP, which is proprietary to windows operating system, is not accessible and is not possible to the manipulate the packets in any of the TCP/IP protocol suite from levels above the TCP/IP driver layer. This makes the use of these techniques in windows operating system more complicated. To overcome this problem, hook technique is developed during this work in order to control the packet at the point that links between the protocol driver and the NIC card(s), which is represented by the Network Driver Interface Specefication (NDIS) .
The work in this paper is, therefore, composed of two parts:

1-NDIS Hook.

2- Data Hiding in IP Packet.

2.1 Network Driver Interface Specefication (NDIS) Hook

(Microsoft Corporation, 2005)

The term hooking represents a fundamental technique that controls a particular piece of code execution. It provides a straight forward mechanism that can easily alter the operating system's behavior as well as third party products, without having their source code available. The proposed system must be run in an user mode. This makes a comfortable and familiar interface to the user for handling routine data and gets the

required information**.** The windows user-mode program can hook the NDIS. This program can monitor, control, add, and modify the NDIS incoming/outgoing packets.

and in order to implement NDIS hook, it must be composed of two parts :

1- NDIS hook driver.
2- NDIS hook application.

The NDIS-Hook driver is logically similar to the NDIS Intermediate (IM) driver, but it is implemented differently. It inserts itself between TCP/IP and all of the adapters that bind with it.

2.2 Data Hiding in IP Packet

This paper assumes that two workstations A and B transfer information overtly over a computer network and employs data hiding via the TCP/IP protocol suite to communicate covert supplementary information.

The Flag field have 3 bits. The first bit is unused and represent random values that the TCP/IP protocol dose not use it[Stallings W 2002].

This paper considers the use of the unused bit as a covert channel that can transfer the proposed data across the network from the sender to the receiver. When this bit are used properly, the Flag field will seem perfectly normal and the hidden data can not be detected by any network monitoring scheme.

A scenario is developed between a sender A who tries to send a covert infomation to a receiver B.

3. Data Hiding Scenario

In this scenario two algorithms are suggested to be implemented by the sender and the receiver. The sender site algorithm includes the data into the packet and sends them to the receiver site. The receiver site algorithm will read the Flag field of the IP packet and verify the covert data from the packet.

3.1 At the Sender Site

If convert information from sender to receiver site, the sender site algorithm will run the following:

1. Check for fragmentation.
2. Read the byte of convert information.
3. Put each bits of the bytes into the first bits of the Flag field of the IP Packet until all bytes is finished.
4. Recalculates IP checksum.
5. Sends the packets.

Afraid of random value in first bit of Flag field, The sender must be send at first time 16 bits that be the signature to make the receiver know that the sender will be send hiding information. After this operator the sender send 16 bits that is the information length to the receiver to know the determine the length of information.

In this paper fragmentation problem are solved by repeated the header in the all fragmented packets, and the sender must check the DF (Do not fragment) if equal 0, the sender will check fragmentation offset if equal 0 then the information will hide normally. If fragmentation offset not equal 0,then the packet will be send without hiding. If DF equal 1 the sender hide information .

The Sender Algorithm

Input: reference sign, length of information, information and
IP packet. (where information = Data)
Output: IP packet containing bit from the covert information.
Step 1 : x = the first least significant bit of the signature.
y = the first bit of length of the covert information.
Signature = length of information= information = false.
i = j = k =0.
Step 2 : get the current sending packet
IF this packet is not destined to go to the specified receiver ***THEN***
Go to step 2
Step 3 *IF* DF=1 (the packet is not fragmented)
Go to step 4
ELSE
IF fragment offset =0 (If the packet is fragmented take the first fragmented packet)
Go to step 4
ELSE
Go to step 2
Step 4 : ***IF*** signature = false ***THEN***
{
put x into the most significant bit of the Flag field.
i = i +1
IF i < 16 ***THEN***
x = the next bit of the signature.
ELSE
{
signature = true
i = 0
}
Step 5 : ***IF*** length of information = false and signature = true ***THEN***
{
put y into the most significant bit of the Flag field.
j = j +1
IF j < 16 ***THEN***
y = the next bit of the length.
ELSE
{
length of information = true.
J = 0.
}
}
Step 6: ***IF*** information = false ***THEN***
{
z = information [i].

m = the first bit of the z.
information = true.
k = 0.
}
put z into the most significant bit of the Flag field.
k = k +1.
IF k < 8 ***THEN***
m = the next bit of z.
ELSE
If k < length then
{
k= k + 1.
information = false.
}
Step 7: recalculate the IP checksum.
Step 8: end.

3.2 At The Receiver Site

On the other hand, the algorithm of the receiver site will be run in the following sequence:

1- First check if there fragmentation.
2- The receiver will always read the first bit of the Flag field.
3- The receiver will read the length of the covert information if it receives a reference sign.
4- The information itself will be read after reading its length and the receiver will wait for another reference sign to arrive.

Since the receiver exists in a network, it may receive packets from many computers. The receiver will first read the source addresses of the packets in order to recognize the specified sender, and then will read the First bit of the Flag field of the sender to extract the covert information. Figure 3-4 shows the behavior of the receiver.

There were a problem is Fragmentation. The header will be repeated in all fragment. To resolve this problem, the receiver must check the DF (Do not fragment) if equal 0, the receiver must check fragmentation offset if equal 0 take information normally. If fragmentation offset not equal 0, let packet go. If DF equal 1 hiding information.

The Receiver Algorithm

Input: IP packet.
Output: the covert information extracting from the IP packet.
Step1 : i= first bit of the signature
M = 0
signature , length of information, information = false .
Step 2 : get packet.
Step 3 *IF* DF=1 *THEN*
Go to step 4
ELSE

IF fragment offset =0 *THEN*
Go to step 4
ELSE
Go to step 2

Step 4 : ***IF*** signature = false ***THEN***
{
k = the most significant bit of the Flag field
IF i = k ***THEN***
i = the next bit of the signature
ELSE
i = first bit of the signature
IF i = the end of the signature ***THEN***
signature = true
}
Step 5 : ***IF*** length of information = false ***AND*** signature = true ***THEN***
{
L = the most significant bit of the Flag field
put l into buffer
IF m = length ***THEN***
length = true
ELSE
m = m + L
}
Step 6: ***IF*** information = false ***AND*** length of information = true ***THEN***
{
put the most significant bit of the Flag field into buffer
j = j+1
IF j = 8 ***THEN***
{
put the buffer value into information[y]
y = y + 1
j = 0
}
IF y = length ***THEN***
{
information = true
goto step 8
}
}
Step 7: go to step 2
Step 8: end

4. Conclusions

This work uses reliably for the first time the last bit of the flag field for transmission of covert information. This bit is unused and has no role in sending any information, besides that there uses do not interfere with any value of the fields in the IP header. In

other words, chasing the values of other fields does not lead to the detection of the hidden data throughout using this bit.

Although there are no other workers that used this bit to compare their results with that observed in the present work, nevertheless, the transmission of covert information can be compared with the result of Kamran who used the Don't Fragment (DF) bit [(K. Ahsan and D.Kundur, 2002).]. Unlike the present work, inspectors can notice peculiar value of the DF bit due to the difference of the identification field in the packets stream.

Furthermore, the length of covert information is unlimited in this work. The larger the length of covert information, the larger the number of packet is required for transmission. Unlike other workers who used ID field, they were only able to transmit only a limited character per a message.

[R.M. Goudar, Prashant N. Patil, (2012) , Llamas D,UK, (2004)].

The following points are concluded throughout this work:

1- Unused bit are exploited during the manipulation of packet header of the IP protocol suite in order to identify the covert channel.
2- The arisen problem by windows XP architecture during covert channel development is solved by the use of NDIS hooking filter driver.
3- The NDIS can control and modify the packet at the application level through hooking process.
4- The similarity between packets containing covert information and normal ones do not lead to detect the covert information by intrusions.
5- Adding supplementary information to the packet without affecting the packet size is possible.
6- When the size of the data is very large fragment must be happen (in any host or router) .

5. Future Work Suggestions:

Covert channels find important applications in many fields; therefore, the future work may be directed in the following manner:

1- Using covert timing channel to transfer the covert information.
2- Analyzing the IPv6 header to hide data in it.
3- Inclusion of a new covert channel technique associated to the manipulation of the header fields into another network protocol suite such as UDP, OSPF, RIP….etc.
4- When the sender send the packet to the receiver it may be dropped by any router on the path, this problem must be solve to avoid losing any data.

Reference

R.M. Goudar, Prashant N. Patil, Aniket G.. Meshram*, Sanyog M. Yewale, Abhay V. FegadeComputer Engineering Department, Pune University, MAE, Alandi Pune, Maharashtra 412105, India **Secure Data Transmission by using Steganography**, 2012

Radu Ciobanu, Ovidiu Tirsa, Raluca Lupu, Sonia Stan, "**Steganography and Cryptography Over Network Protocols"**,2011.

Miss D. D. DhobaJe Dr. V. R. Ghorpade Mr. B. S. Patjj Mrs. S. B. Patil **"Steganography By Hiding Data In Tcp/Ip "**Headers",2010.

K. Ahsan and D. Kundur, **Practical data hiding in TCP/IP**, Proc. ACM Workshop on Multimedia Security, 2002.

D Kanh, **The History of Steganography, Proceedings**: Information Hiding. First International Workshop, Cambridge UK pp1-5 2008.

Enrique Cauich1, Roberto Gómez 2, Ryouske Watanabe2 **Data Hiding in Identification and Offset IP fields** California University at Irwing, Computer Science and Engineering 204B University of California, Irvine, CA 92717 , 2004.

McHugh J.," **Covert Channel Analysis**", Handbook for the Computer Security Certification of Trusted System, 2005.

M. Schaefer, B. Gold, R. Linde, and J. Scheid, **"Program confinement in kvm/370,"** Annual ACM Conference, October 2004.

J. C. Huskamp, **Covert Channels in Timesharing System**. PhD thesis, University of California, Berkeley, California, 2008.

R. A. Kemmerer**, "Shared resource matrix methodology: An approach to identifying storage and timing channels,"** vol. 1 of 3, pp. 256–277, ACM Transactions on Computer Systems, August 2007.

Jessica Fridrich and Miroslav **Goljan, Practical steganalysis of digital images**: state of the art, Proceedings of SPIE -- Volume 4675 Security and Watermarking of Multimedia Contents IV, April 2002, pp. 1-13

Chandramouli,R.; Subbalakshmi, K.P**., Active steganalysis of spread spectrum image steganography Circuits and Systems,** 2003. ISCAS '03. Proceedings of the 2003 International Symposium on , Volume: 3, May 25-28, 2003

Buetler, I. (2008). **Covert Channel Attacks – Inside-out Attacks, Compass Security Retrieved from:**
www.csnc.ch/misc/files/publications/covert_channel_csnc_v1.0.pdf

Stallings W., **"Data and Computer Communications"** 5th edition, ISBN 0-13-571274-2, New Jersy Prentice-Hall Inc, 2002

Microsoft, **"NDIS - Network Driver Interface Specification",** Microsoft Corporation, 2005. http://www.microsoft.com/whdc/device/network/ndis/default.mspx

Llamas D.," **Covert Channel Analysis and Detection with a Reverse Proxy Servers using Microsoft Windows**", School of Computing, Napier University, EH10 5DT, Scotland, UK, 2004