A Network Covert Channel Based on Packet Classification

Ping Dong, Huanyan Qian, Zhongjun Lu, and Shaohua Lan
(Corresponding author: Ping Dong)

School of Computer Science and Technology, Nanjing University of Science and Technology
200 Xiao Lig Wei Street, Nanjing, P. R. China
(Email: dongping327@gmail.com)

(Received Nov. 23, 2010; revised and accepted Feb. 28, 2011)

Abstract

A network covert channel violates security rules and leaks information imperceptibly. Many researchers have been studying how to construct it, and the basic methods involve exploiting packets head, packets sorting and packets timing, etc. To construct a more secret covert channel, this paper proposes a novel approach based on packet classification. With this method, covert information is encoded by modulating the varieties of packets on the Internet. The basic idea of packet classification, choice of carrier and algorithm of information hiding based on packet classification are discussed. Our analysis demonstrate that the covert channels based on packet classification can not be totally eliminated with current methods.

Keywords: Information security, network covert channel, packet classification

1 Introduction

Covert channels in computer network protocols violate the systems security policy and leak information imperceptibly [9]. Essentially, they can be used to transfer information to users or system processes that normally would not be allowed to access the information. They also can be combined with Trojan horses and make a serious threat to network information security, which can not be conquered by traditional security technologies based on encryption.

A covert channel, to be useful, does not need high bit rate or high capacity or even low loss rate. It is generally satisfactory if it can transmit a few bits per second with some positive probability. For example, only a few bits are needed to disclose the PIN number of a personal bank account. However, a covert channel, to be effective, must be difficult to detect. This is a paramount requirement. So the goal of this work is to build a channel which is, first of all, stealthy, and more, has as much capacity as possible. In this paper, we present a novel approach based on packets classification to construct covert channels, which makes use of many fields of multi-protocol, even can combine with the timing feature of covert channel. This covert channel is difficult to be detected by wireshark or IPS/IDP, as the packets it takes along the convert information imitate the normal network packets. More details will be elaborated later.

In Steganographic systems, there are always three factors competing: bandwidth vs robustness vs stealthiness. By increasing the bandwidth, the channel used to hide information becomes more susceptible to noise and being discovered by adversaries. Lowering the hidden bandwidth causes the transmission to be stealthier and more resistant to change due to external noise. We try to find a compromise among these factors.

The rest of this paper is organized as follows. In Section 2, we start with a brief description of related work in covert channels. In Section 3, we provide the basic idea of covert channel based on packets classification to introduce the specific method of packets classification. The model of covert channel based on packets classification, including particular algorithm of encode and decode of information hiding, is then described in Section 4. In Section 5, we provide some improvements to enhance the stealthiness of proposed channel. Section 6 presents the performance analysis. Finally, in Section 7, we draw some conclusions and point to future work.

2 Related Work

Many researchers have been studying how to construct network covert channels and relevant countermeasures respectively, since the channels were introduced [10, 13]. In general, covert channels can be classified into two categories: storage-based and timing-based. In storage-based covert channel, the messages are usually embedded into protocol header fields, including unused header bits, header extensions and padding fields. And these covert channels use a variety of protocols including TCP, IP, HTTP, FTP, and DNS [4, 6, 7, 8, 19]. These approaches are easy to realize and have much high bandwidth. However, they are vulnerable to active defence systems [18]. A timing-based covert channel, on the other hand, relays covert messages based on the timing relationship of the packets, such as packet timing, packet loss and packet sorting, etc. [3, 5]. It is hard to detect since those timing features are not directly perceived through the senses. But the shortcomings are complicated control management and low-bandwidth. Recently, several varieties of exploits within the various network protocols are revealed, including some ways based on the third part resources. Examples include [1, 12, 15]. Those approaches have even better stealthiness, but the channel noise cannot be eliminated completely.

Most of the past studies on covert channels have been concentrated on wired computer networks, there are also some experiments applying to wireless communication, like ad-hoc wireless networks. Literature [11] investigates ad-hoc wireless networks' susceptibility to covert channels that can be formed through manipulating the network protocols, and verifies these covert channels are very difficult to eliminate or even detect.

In contrast to aforementioned covert channels, we take special care to the classification of network packets in order to combine the storage features with timing features to a certain degree. Thus we can get a much higher bandwidth channel on the basis of stealthiness.

3 Method of Packets Classification

In TCP/IP, the information is encapsulated in the IP packet. The basic unit of message transfer on IP layer is IP packet, so we can abstract the communications in TCP/IP as the transfer of IP packet. The packet we mentioned in the article is IP packet. In order to construct a universal model of covert channels, based on IP layer, we abstract the communication on IP layer as channel, IP packet as carrier (IP packet on the channel is not divided), so the secret information can be embedded in the carrier. The whole process of covert messaging is as follows: based on the IP packet, the sender abstracts the classifiable features from the different features of IP packet, and then the information can be mapped into the classifiable features. After delivery on the Internet, the recipient just needs to identify the features and revert to the original message.

Packets classification can be done on different hierarchies. There are basically three classes. Class A is distinguished by protocol, for example, the Network Layer protocol (ICMP, IGMP), the Transport Layer protocol (TCP, UDP), the Session Layer protocol (SSL), and the Application Layer protocol (DNS, HTTP, SMTP, and FTP). Class B is distinguished by packet header, including the different values of some fields and the very existence of some fields. These fields cannot be written secret message by themselves, because the protocol has already regulated the values, but we can modulate the fields as features for covert communication. The values of these fields completely keep to the RFC rules, so they are not suspicious. The connection request packet and connection accepted packet of TCP, ICMP are all useable. Class C is distinguished by unused header bits, header extension and padding fields. This is a special form of packet classification. Those fields are redundant, so covert information can be encoded in them directly when the detect system doesn't care about them. The extension, option, unused fields, padding, and preserving fields can all be used to transmit covert data. In conclusion, the vast number of different protocols in the Internet and variety of header fields makes the packet classification complicated, and the number of packets provided with different features is huge.

In order to describe the packets classification clearly, we introduce line digraph. Figure 1 shows the elementary classification based on class A.

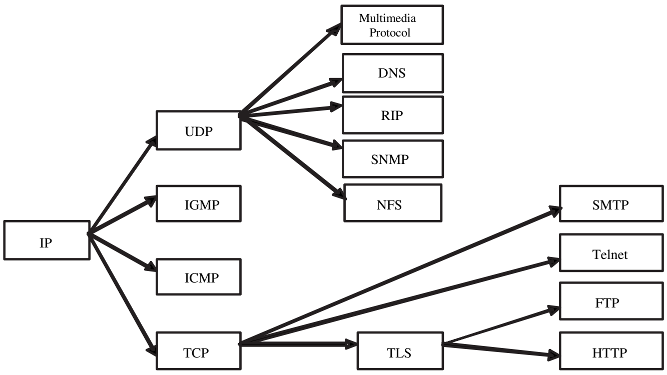


Figure 1: The line digraph of the elementary classification based on class A

As shown in Figure 1, each node is a Protocol Data Unit (PDU), which is a feature. All the notes from start to end on every path will compose a completely whole IP packet. There are eleven paths in Figure 1, that is to say, eleven classified packets exist on class A level.

But Figure 1 is unshaped; we can refine it along one of the paths and get a more specific classification. On class B, from IP to HTTP, we get a refinement as Figure 2.

In Figure 2, each note is a feature of PDU. These fields are required and the values of the features are already regulated by RFC. Figure 2 is also not precise yet. We only utilize IP head and TCP head and the path can be classified more specifically.

Class C is distinguished by unused header bits, header extensions and padding fields. The fields can be written whatever you want. But we can see that, because of design limitations, these fields can be used for covert communication easily and also are prone to be detected and attacked.

All features we mentioned above are about the value of packet head fields, there are some other timing features,

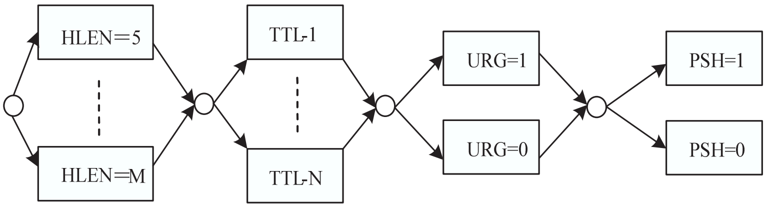


Figure 2: The line digraph of the elementary classification based on class B

like packet rates, can be encoded. Based on the theory of packet classification, storage channels and timing channels both use packet features to transfer covert messages, so data features and timing features have no differences essentially, just the method of hide and recover message are diverse. If data features and timing features are combined together, we will get a mixed classification method. A universal model that can unite storage channels and timing channels will be established. The channels have higher bandwidth and more stealthiness, but corresponding channel control is much more complicated. Since our focus is on storage channels, we will not further discuss timing channel.

4 Model of Covert Channel Based on Packets Classification

4.1 Channel Carrier Selection

Choosing an appropriate carrier is the first step to construct a covert channel. Theoretically, a packet can be a carrier as long as it has the features that can be recognized and modulated. But the channels capabilities are considerably diverse when different carriers are chosen. The packets categories are numerous, and we should ensure that all the packet features we use must meet the following points:

- It should be modulated in legitimate scale and not violate the rules;
- It should be transferred at any time and not disobey the session rules;
- It should be sent in any order since some packets with features sending in special sequence may bring suspiciousness.

Generally speaking, the channel that communication system supports is a binary channel. Binary channel is the most common channel, which has two recognized features. We assign two features as 0 and 1. Binary channel can hide very few messages, so its invisibility is relatively good, and the detection software based on statistics can hardly find it. But its main shortcoming is low-bandwidth. In packets classification, features in one packet are quite a lot, and one packet can be embedded with many bits, so non-binary channel is considered better. Its bandwidth is high, and it is relatively not easy to be detected. In summary, non-binary channel is main carrier of covert communication based on packets classification.

4.2 Encoding Algorithm of Information Hiding

The modulation of binary channel carrier is simple, which is mapped one by one, while the modulation of non-binary channel carrier is comparatively complicated. Mapping encoded characters of hidden information to features of packets is divided into the following two steps: mapping encoded characters of hidden information to sorting vector and then mapping sorting vector to features of packets.

In order to map encoded characters of hidden information to sorting vector, a sorting vector based algorithm (SVB) is generated. Firstly, secret information should be transformed into bits stream. Secondly, the bits stream is partitioned into several sub-groups according to feature numbers t. Let Num be the unique unsigned integer of each sub-group, which is input of SVB.

As mentioned previously, one packet can be classified as a variety of state by A class hierarchy, then the state can be divided into k different features in accordance with B class hierarchy. The features can be expressed with x_1, x_2, x_3, …, x_k. Supposing the number of possible legal features of x_i is m_i, $i \in [l, l]$, a vector W can be expressed as follows:

$$W = (x_1, x_2, x_3, \ldots, x_k). \qquad (1)$$

x_i is a positive integer, and $x_i \in [l, m]$. The amount of W that meets this condition should be $\prod_{i=1}^{k} m_i$. All the vectors can be numbered 0, 1, 2, … with a certain sequence, and the maximum bits of the serial number, which represented by L, can be expressed as follows:

$$L = \lfloor \log_2(\prod_{i=1}^{k} m_i) \rfloor = \lfloor \sum_{i=1}^{k} \log_2 m_i \rfloor \geq t. \qquad (2)$$

So, the range of Num is equal to or falls within the range of serial numbers of W.

According to this principle, we can map Num to the serial numbers of W one by one. There's a rule of how sorts below:

- Horizontal orientation: comparing from left to right, if $i < j$, then x_i has higher priority than x_j.
- Vertical orientation: comparing in one vector component x_i, the vector that has smaller vector component value has higher priority, and should be put in front.

In order to assign Num to x_i from left to right, SVB algorithm can be demonstrated as follows. The precondition is $k > 2$, when $k \leq 2$, there's an easier way to map it.

Step 1. Initialization. Let $Num = Num + 1$, $i = 1$, i controls every vector component by looping.

Step 2. Assign the current vector component. If $Num \bmod (\prod_{j=i+1}^{k} m_j) == 0$, then $x_i = Num/\prod_{j=i+1}^{k} m_j$, or else, $x_i = Num/\prod_{j=i+1}^{k} m_j + 1$.

Step 3. $Num = Num - (x_i - 1)\prod_{j=i+1}^{k} m_j$, $i = i + 1$. i points next vector component.

Step 4. If $i == k$, let $x_i = Num$, and the whole algorithm is over; or else, repeat the second step until all vector components are assigned.

When the whole algorithm is over, every vector component will be assigned a value and the map from encoded characters of hidden information to sorting vector is achieved.

SVB algorithm is adapted to $k > 2$. When $k = 1$, the way to map is obvious. When $k = 2$, there's only two features, m_1 and m_2. Using the same idea of map, SVB algorithm can be simplified to be matrix map algorithm. It is a matrix $m_1 * m_2$ Mr, consisting of integers range from 0 to $m_1 * m_2 - 1$ that should be arrayed by row priority which can be expressed as Equation (3), or column priority which can be expressed as Equation (4). Row priority or column priority can be used as a parameter in channel. Each bits group gets a Num, and the row location i and column location j of Num can be found in Mr, so $W = (i, j)$ is generated.

$$Mr = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m_2} \\ a_{21} & a_{22} & \cdots & a_{2m_2} \\ \cdots & \cdots & \cdots & \cdots \\ a_{m_1 1} & a_{m_1 2} & \cdots & a_{m_1 m_2} \end{pmatrix} = \tag{3}$$

$$\begin{pmatrix} 0 & 1 & \cdots & m_1 - 1 \\ m_1 & m_1 + 1 & \cdots & 2m_1 - 1 \\ \cdots & \cdots & \cdots & \cdots \\ m_1 * m_2 - m_1 & m_1 * m_2 - m_1 + 1 & \cdots & m_1 * m_2 - 1 \end{pmatrix}$$

$$Mr = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m_2} \\ a_{21} & a_{22} & \cdots & a_{2m_2} \\ \cdots & \cdots & \cdots & \cdots \\ a_{m_1 1} & a_{m_1 2} & \cdots & a_{m_1 m_2} \end{pmatrix} = \tag{4}$$

$$\begin{pmatrix} 0 & m_1 & \cdots & m_1 * m_2 - m_1 \\ 1 & m_1 + 1 & \cdots & m_1 * m_2 - m_1 + 1 \\ \cdots & \cdots & \cdots & \cdots \\ m_1 - 1 & 2m_1 + 1 & \cdots & m_1 * m_2 - 1 \end{pmatrix}$$

Preceding part achieves the map from encoded characters of hidden information to sorting vector, the next step is mapping sorting vector to features of packets. We establish the correspondence between sorting vector and features of packets, as illustrated in Table 1.

Table 1: Map from sorting vector to features of packets

	Value of vector component
Value component	**1, 2, 3, . . ., m_i**
x_1	$A_{1,1}$ $A_{1,2}$ $A_{1,3}$ . . . A_{1,m_i}
x_2	$A_{2,1}$ $A_{2,2}$ $A_{2,3}$ . . . A_{2,m_i}
x_3	$A_{3,1}$ $A_{3,2}$ $A_{3,3}$ . . . A_{3,m_i}
. . .	. . ., . . ., . . ., . . ., . . .,
x_k	$A_{k,1}$ $A_{k,2}$ $A_{4,3}$ . . . A_{k,m_i}

As shown in Table 1, for each x_i, there is a $A_{i,j}$, which is the specific feature value, when the serial number of feature x_i is j. m_i is the number of features of x_i. Undoubtedly, the specific feature value is correlated with carrier. In most situation, $m_i \neq m_j (i \neq j)$. Through checking the comparison in Table 1, we can construct eligible IP packet which has specific features.

Let us take an example to make the SVB algorithm more comprehensible. Supposing we choose three features, including URG(0,1), PSH(0,1), HLEN(5,6,7,8). The number of each feature is 4, 2, 2, that is, $m_1 = 2$, $m_2 = 2$, $m_3 = 4$. According to Equation (2), $L = 4$. Let t be 4, and the serial number is within [0, 15]. Table 2 shows a sorted vector W based on the rule.

Table 2: An example of sorted vector

No.	**Vector**	**No.**	**Vector**
0	(1, 1, 1)	8	(2, 1, 1)
1	(1, 1, 2)	9	(2, 1, 2)
2	(1, 1, 3)	10	(2, 1, 3)
3	(1, 1, 4)	11	(2, 1, 4)
4	(1, 2, 1)	12	(2, 2, 1)
5	(1, 2, 2)	13	(2, 2, 2)
6	(1, 2, 3)	14	(2, 2, 3)
7	(1, 2, 4)	15	(2, 2, 4)

Suppose we want to transfer "A", whose bits stream is 01000001. Firstly, we divide it into high four bits and low four bits, and we get $Num_1 = 4$, $Num_2 = 1$. Then we calculate the vector W.

Step 1. $Num_1 = Num_1 + 1 = 5$, $i = 1$;

Step 2. $Num_1(\text{mod})(4 * 2) = 5$, $x_1 = 5/8 + 1 = 1$;

Step 3. $Num_1 = Num_1 - (x_1 - 1) * 8 = 5$, $i = 2$;

Step 4. $Num_1(\text{mod})4 = 1$, $x_2 = 5/4 + 1 = 2$;

Step 5. $Num_1 = Num_1 - (x_2 - 1) * 4 = 1$, $i = 3$;

Step 6. $x_3 = Num_1 = 1$.

So we get $W = (1, 2, 1)$ when $Num_1 = 4$. In the same way, we can get $W = (1, 1, 2)$ when $Num_2 = 1$. When we check the comparison in Table 3, we can construct eligible IP packet. The two packets have the features (URG=0, PSH=1, HLEN=5) and (URG=0, PSH=0, HLEN=6) respectively. In Table 3, $*$ means there's no feature available.

4.3 Decoding Algorithm of Information Hiding

Recovering the message is the reverse process of embedding it, which is also divided into two steps: mapping features of packets to sorting vector and then mapping sorting vector to encoded characters of hidden information.

When a packet arrives, we can get the relative vector component W directly by checking the table of map between sorting vector and features of packets (see Table 1). Then a corresponding algorithm of SVB is designed to get Num by following expressions:

$$Num = \sum_{j=2}^{k}\{(x_{(j-1)-1}) * \prod_{i=j}^{k} m_i\} + x_k - 1. \quad (5)$$

Num, output of the algorithm, is the bits stream of each message group. We just need to combine and translate them, and the secret information is reverted.

5 Improvements

A stealthy covert channel not only needs to obey the RFC rules, but also have a coincidence with normal network traffic characteristic. In order to avoid detection by network protocol analyser, which is based on abundant data analysis, randomizing and homogenizing the packet types may be a good method, which can prevent singleness of packets with some protocols from bringing exceptional statistic.

Randomizing and homogenizing packet types should be able to perform two main functionalities, namely:

- Utilize redundant packet types in order to ensure the same chars can match different packet types;
- According to the analysis of existing network packets, randomize and homogenize packet types to imitate regular pattern.

If we just take two features T_i and T_j from the packet feature set C_N, there will be two types. Now we take a sub set C_{sub}, $C_{sub} = \{T_1, T_2, \ldots, T_r, T_{r+1}, \ldots, T_{2r}\}$, where r is a positive integer.

$$rd(x) = \begin{cases} random()\%r + 1, & \text{x=type}_1; \\ random()\%r + r + 1, & \text{x=type}_2. \end{cases}$$

Where $random()$ is a function that can generate random integers by the pseudo-random number algorithms, and it is typically used in computer programs. $rd(x)$ is a uniformly distributed random variable in the $[1, r]$ and $[r + 1, 2r]$ range. Figure 3 shows the contrast between the non-randomized and randomized packets.

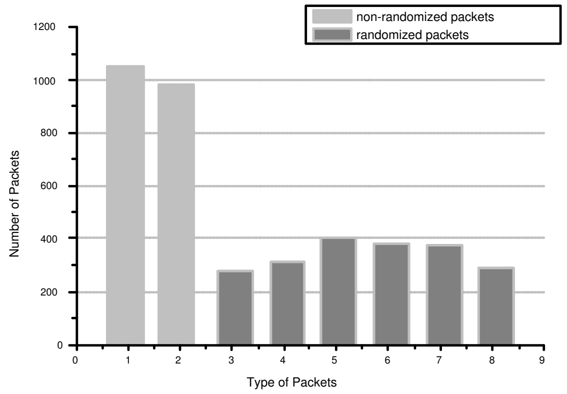


Figure 3: The line digraph of the classification based on one path of class B

In Figure 3, the left two columns are non-randomized packets, and the right six columns are randomized packets where $r = 3$ in the random function. Obviously, non-randomized packets have limited quantities and arouse anomalous traffic while randomized packets average the flow to many kinds of packets, which is much more approached to the normal traffic. The value of r is bigger, the effectiveness of randomization is better. However, that needs more packet features those will cause over-complicated control management. So we have to consider the trade-off.

6 Experimental Results

6.1 Implementation Details

In our implementation, we send fabricated packets over IP using WinPcap [16]. WinPcap is the industry-standard tool for link-layer network access in Windows environments: it allows applications to capture and transmit network packets bypassing the protocol stack, and has additional useful features, including kernel-level packet filtering, a network statistics engine and support for remote packet capture.

We select ICMP packets as the main experiment object. ICMP packets play such a crucial role in IP packet management, and not rely on special ports, so they are favorable to the proposed channel.

The format of an ICMP message is shown above. We take Type and Code field as two features. Table 4 shows frequently-used ICMP packets that can be delivered by

Table 3: An example of map from sorting vector to features of packets

	Value of vector component			
Value component	**1**	**2**	**3**	**4**
x_1	URG=0	URG=1	*	*
x_2	PSH=0	PSH=1	*	*
x_3	HLEN=5	HLEN=6	HLEN=7	HLEN=8

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |     Code      |          Checksum             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             unused                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Internet Header + 64 bits of Original Data Datagram      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4: Format of ICMP message

host machine. Echo message, echo reply message, timestamp message and timestamp reply message have nonzero value, so we take these four types as available features, that is (8,0), (0,0), (13,0), (14,0). Detailed encoding and decoding procedures can find in Section 4, in this part, we just analyse the experimental results.

Table 4: Frequently-used ICMP packets

Type	Code	Message Type
3	2	protocol unreachable
3	3	port unreachable
11	1	fragment reassembly time exceeded
12	1	Parameter Problem
13	0	timestamp message
14	0	timestamp reply message
8	0	echo message
0	0	echo reply message

6.2 Stealthiness Analysis

According to the existing literature, real network traffic submits to Poisson distribution and some similar distributions [2]. Literature [17] proposes a covert channel detective approach based on network traffic analysis, and provides the detective rules when real network traffic submits to Poisson distribution. In this article, we also consider that real network traffic submits to Poisson distribution.

We send random message and observe the channel 10 minutes, and we can get the distribution showed in Figure 5. We use SPSS (Statistical Product and Service Solutions) [14] to analyse these packets of four types. SPSS Statistics is the world's number one choice for reliable statistical analysis. It is a comprehensive, easy-to-use set of data and predictive analytics tools for business users, analysts and statistical programmers. Figure 6 shows that the packets of four types all submit to Normal distribution to some extent. According to the theory of probability statistics, the Poisson (λ) distribution is approximately normal $N(\lambda, \lambda)$ for large values of λ. We consider that the proposed channel traffic does not strictly fit Poisson distribution, but it is much close with long time transmission. So this channel is stealthy and can resist detection to a certain degree.

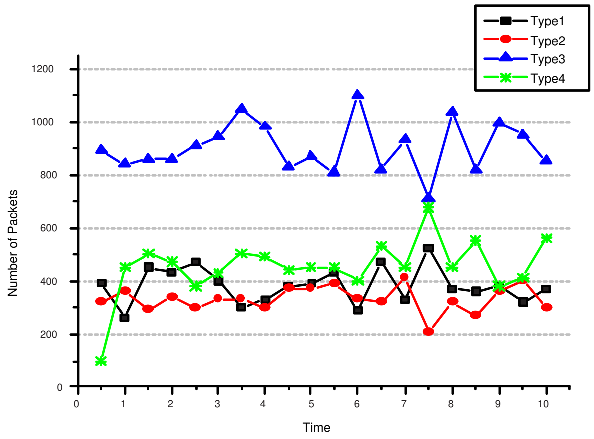
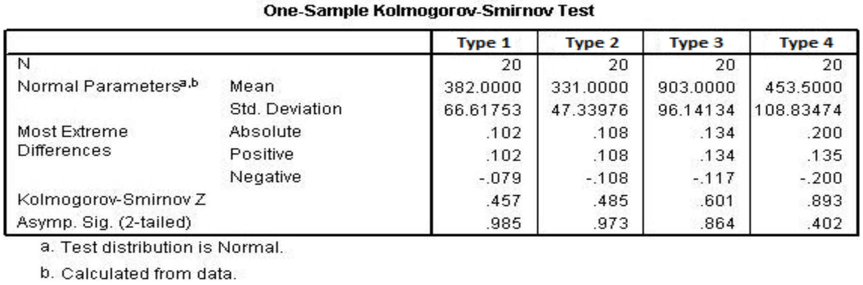


Figure 5: Number of packets under increasing time

In addition, according to the theory of channel and basic idea of packets classification, covert channels should not be eliminated completely unless the channel carrier, say, modulation of the packets, can be eliminated. In TCP/IP, there are abundant protocols to satisfy different application requirement. These protocols have various features in their heads, which is necessary to ensure success of communication. TCP/IP suit is a hierarchical model, and it is impossible to eliminate the features by flatting it. Apart from that, the crucial applications have already relied on TCP/IP, and eliminating the features is equal to design Internet afresh. Of course, it is not worth. So network covert channel based on packets classification can be inevitable.



One-Sample Kolmogorov-Smirnov Test

		Type 1	Type 2	Type 3	Type 4
N		20	20	20	20
Normal Parameters[a,b]	Mean	382.0000	331.0000	903.0000	453.5000
	Std. Deviation	66.61753	47.33976	96.14134	108.83474
Most Extreme Differences	Absolute	.102	.108	.134	.200
	Positive	.102	.108	.134	.135
	Negative	-.079	-.108	-.117	-.200
Kolmogorov-Smirnov Z		.457	.485	.601	.893
Asymp. Sig. (2-tailed)		.985	.973	.864	.402

a. Test distribution is Normal.

b. Calculated from data.

Figure 6: Distribution test (SPSS)

6.3 Capacity Calculations

The bandwidth of this covert channel is determined by the features of packets we use. The bit rate is approximately $\lfloor \log_2 t \rfloor$ bits/packet, where t is the number of packet features. Only in Figure 2, we can get $\lfloor \log_2^{(m-4)*N*2*2} \rfloor$ bits/packet, where M and N are positive integer, and $M \in [5, 16]$, $N \in (1, 255]$. The more features we use, the more secret information can be embedded. If the packet features of all three classes can be used adequately, there will be tremendous channel capability. But we have to consider the stealthiness and control complexity.

7 Conclusions and Future Work

In this paper, we propose a design for a network covert channel based on packets classification. By modulating the varieties of packets on the Internet as carrier of covert information rationally, this channel can deliver tremendous messages. According to the analysis, this steganography is valid. Randomizing the packets can avoid the detection of common network protocol analyzer effectively. However, in practical channel, except capacity and stealthiness, we also have to consider channel management, including error detection and relevant control process. So some built-in error detection/correction capabilities are needed, in this way, our channels have the ability to be extremely robust and resilient to external disturb effects.

Moreover, the trade-off among bandwidth, robustness and stealthiness has to research much more. Future work will focus on finding a compromise among these factors and analysing the consequences of changes applied to each one.

In addition, the protocols in TCP/IP are plentiful, available packets are abounding. How to dig more packets features for carrier, especially the timing features of packets, is also an interesting spot. It would also be useful to develop a covert channel model with good performances.

References

[1] L. Bai, Y. Feng Huang, G. N. Hou, and B. Xiao, "Covert channels based on jitter field of the RTCP header," *International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 1388-1391, 2008.

[2] M. Chen, L. Pei, and W. Liang, "Double-mode model about distributions of network traffic," *Journal on Communications*, vol. 29, no. 5, pp. 100-106, 2008.

[3] A. E. Atawy, and E. A. Shaer, "Building covert channels over the packet reordering phenomenon," *Proceedings of INFOCOM - The 28th Conference on Computer Communications*, pp. 2186-2194, 2009.

[4] T. M. Gil, *IP-over-DNS using NSTX.* (http://thomer.com/howtos/nstx/)

[5] J. Giles and B. Hajek, "An information-theoretic and gametheoretic study of timing channels," *IEEE Transactions on Information Theory*, pp. 2455-2477, 2002.

[6] T. Handel and M. Sandford, "Hiding data in the OSI network model," *Proceedings 1st International Workshop Information Hiding*, pp. 23-38, 1996.

[7] A. Hintz, *Covert Channels in TCP and IP Headers.* (http://www.def.con.org/images/defcon-10/dc-10-presentations/dc10-hintz-covert.ppt)

[8] Z. Kwecka, *Application Layer Covert Channel Analysis and Detection*, Technical Report, Napier University Edinburgh. (http://www.buchananweb.com.uk/zk.pdf)

[9] B. Lampson, "A note on the confinement problem," *Proceedings of the Communications of the ACM*, vol. 16, no. 10, pp. 613-615, 1973.

[10] S. Lander, G. Armitage, and P. Branch, "A survey of covert channels and countermeasures in computer network protocols," *IEEE Communications Surveys and Tutorials*, vol. 9, no. 3, pp. 44-57, 2007.

[11] S. Li and A. Ephremides, "Covert channels in ad-hoc wireless networks," *Ad Hoc Networks*, pp. 135-147, 2010.

[12] X. Luo, Edmond W. W. Chan, and Rocky K. C. Chang, "CLACK: A network covert channel based on partial acknowledgment encoding," *IEEE International Conference on Communications*, pp. 1-5, 2009.

[13] A. Patel, etc., "Covert channel forensics on the internet: Issues, approaches, and experiences," *International Journal of Network Security*, vol. 5, no. 1, pp. 41-50, July 2007.

[14] SPSS. (http://www.spss.com/)

[15] H. Wentao, Zhi Xuw, and N. Chen, "Research of Covert channels based on web counters," *Journal of Shanghai Jiaotong University*, vol. 42, no.1, pp. 1678-81, 2008.

[16] WinPcap. (http://www.winpcap.org/)

[17] J. Xue, S. Xu, and X. Wu, "A network covert channel detecting model based on traffic analysis," *Computer Engineering*, vol. 28, no. 12, pp. 46-48, 2002.

[18] C. Zhiyong, Y. Qu, F. Li, and C. X. Shen, "Enumerative covert channel audit model in MLS networks," *9th International Conference on Signal Processing*, pp. 2964-2967, 2008.

[19] X. Zou. etc., "The research on information hiding based on command sequence of FTP protocol," *Proceedings 9th International Conference on Knowledge-Based Intelligent Information and Engineering Systems*, pp. 79-85, 2005.

Ping Dong received her B.S. degree from Henan University of Finance and Economics in 2008, and now she is a Ph.D. candidate in Nanjing University of Science and Technology, all major in Computer Science. Her research interests include network security, privacy and anonymity, and wireless network design.

Huanyan Qian is a professor of Nanjing University of Science and Technology. He is executive director of computer users association in Jiangsu province and he is also vice-chairman of higher school education technology seminar of Jiangsu province. He has published 16 textbooks books on methods of compiling, and methods of computing, etc. He is the author of more than 80 scientific papers. His research interests include Network application technology, network security, and IPv6 technology.

Zhongjun Lu received his M.S. degree from Computer Science Department of Nanjing University of Science and Technology. He currently works in ZTE Communication Company, one of most famous communication device providers in China. His research direction is Network Communication and Information Security.

Shaohua Lan received his B.S. degree in the Department of automatic control in 1982, and M.A. and Ph.D. degrees in the Department of Computer Science in 1987 and 2003 separately, all from Nanjing University of Science and Technology (NUST). He has published three books on TCP/IP protocol technology, computer network security fields. He has won several provincial and national awards. He is the author of more than 30 scientific papers in wireless network and network security. He is current on the duty of several projects on wireless multi-hop network and industry control network. His research interests are network security and wireless network design.