A Perceptual Hash-based Approach to Detect Covert Timing Channels

Linfan Wang and Yonghong Chen
(Corresponding author: Yonghong Chen)

College of Computer Science, Technology, Huaqiao University
No.668 Jimei Avenue, Xiamen, Fujian, China 361021
(Email: iamcyh@hqu.edu.cn)

(Received Apr. 12, 2019; Revised and Accepted Oct. 4, 2019; First Online Dec. 7, 2019)

Abstract

Covert timing channels have received intensive interests in recent years, which are integrated into the existing resources of network systems. This means that the traditional security policy, such as firewalls and intrusion detection system, can not capture them effectively. However, there is still not a generic mechanism used to detect a large variety of covert timing channels. Thus, it is a challenging task to detect and disrupt them. In this paper, we introduce a perceptual hash-based approach to detect covert communications. The proposed approach utilizes the extracted perceptual features from the network traffic and the designed perceptual hash functions to classify the traffic as covert or overt. We extracted the perceptual features from four typical and different covert timing channels and tested each of them independently. The experimental results verify that the perceptual hash-based approach is effective and prove that the perceptual hash technique has great potential for blind detection of covert timing channels.

Keywords: Covert Timing Channels; Network Security; Perceptual Features; Perceptual Hash

1 Introduction

Covert timing channels (CTCs) transmit the covert information using the elements of existing network resources that were not designed for communication. This makes them to evade the detection of network security mechanisms like firewalls and intrusion detection systems (IDS) [5,16,25]. By manipulating the time or ordering of network events (*e.g.*, data packets), CTCs transmit the secret information from a network system with higher security privilege to the Internet. A scenario where CTCs are used is illustrated in Figure 1. Unfortunately, CTCs are used for harmful intentions and that are the ways of damaging network security, which proposes a challenging task to detect a large variety of CTCs.

There are many existing detection methods for CTCs, while these methods have certain limitations. Some detection methods [1, 2, 9] only detect a specific CTCs algorithm which lack a common mechanism. The other methods [7,17,22] could detect most of CTCs algorithms. However, due to the high-speed network environment, these detection methods lack certain detection robustness to capture CTCs. In summary, the existing detection methods for detecting CTCs has some disadvantages in a high-speed network environments. The details will be described in Section 2.

For the shortcomings of existing detection methods above, we introduce a perceptual hash-based approach to detect CTCs. Perceptual hash [6,15,23] is a new technique in multimedia information security. It solves the problem that the traditional hash doesn't have perceptual robustness. Perceptual hash technique is commonly used in fields such as the image [10,19,24] and voice authentication [14,20,26], while has never been applied into the detection of CTCs in network systems. We observe that the extracted perceptual features of CTCs are different from the features of the original process, but there is a striking similarity among the CTCs features of the same class. The similarity and discrimination of perceptual features give new ideas for the detection of CTCs. Therefore, we have studied the way of appling the robustness and discrimination of perceptual hash into the detection of CTCs.

More specifically, we convert the network traffic into the corresponding time series. The discrete wavelet transform (DWT) is designed to extract the perceptual features of time series in frequency-domain, and the information entropy is designed to extract the perceptual features of time series in time-domain. The perceptual features are then transformed into the perceptual sequences by the designed perceptual hash functions. The purpose is to preserve the similarity and discrimination of the perceptual features and reduce data amount by the robustness and summary of perceptual hash. We last perform the detection threshold estimation of perceptual sequences for both covert and legitimate traffic, and determine whether

the covert traffic is accurately detected. To evaluate the proposed approach, we conducted a series of experiments to verify whether it is effective to detect multiple CTCs. The experimental results show that the perceptual hash-based approach has solved the following problems. In the case of samll sample size, the true-positive rates of CTCs is improved by the good discrimination of perceptual features. At the same time, the proposed approach has a good detection robustness compared with other approaches mentioned in this paper in the case of network interference, such as jitter, packets loss, *etc.*

The rest of this paper is structured as follows: Section 2 introduces the background, related works and existing detection approaches of CTCs. In Section 3, the principle and algorithm of the perceptual hash-based approach are described. Section 4 passes the detection experiments of legitimates and four typical CTCs to validate the effectiveness of the proposed approach in this paper. The content of this paper is summarized and our future work direction is discussed in Section 5.

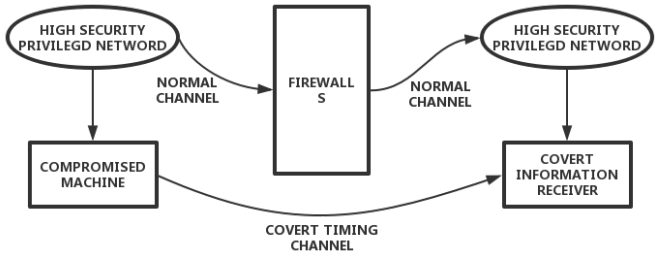

Figure 1: CTCs scenario

2 Background and Related Works

2.1 Threat Analysis of CTCs

There are two classic kinds of CTCs: Active and passive. Active requires communication to generate a additional traffic that is not designed in the network system. Passive transmits the covert information by manipulating the existing traffic in the network system. A scenario in which active and passive CTCs are used is shown in Figure 2. In order to better detect active and passive CTCs, we chose two typical active CTCs, IP - Covert Timing Channel (IPCTC), Time-Replay Covert Timing Channel (TRCTC) and two typical passive CTCs, Distribution-Matching Covert Timing Channel (DMCTC), JitterBug to test the reliability of our approach. The details are introduced in the next sections.

2.1.1 IPCTC

Cabuk *et al.* [3] designed the first IPCTC that is a covert timing channel operating at the IP layer. IPCTC encodes a 1-bit by transmitting a data packet during a inter-packet delay t and encodes a 0-bit by not transmitting packets during a inter-packet delay t. The receiver receives

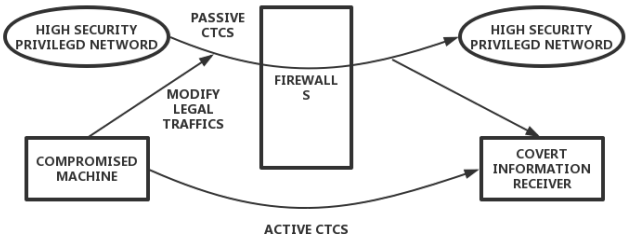

Figure 2: Active and passive CTCs scenario

the size of inter-packet delays to recover the covert information. For the multi-band channels, IPCTC uses the multiple inter-packet delays, each inter-packet delay corresponding to one code. The characteristic of IPCTC is that if the inter-packet delay t is set to a fixed value, the distribution of IPCTC would be close to the geometirc distribution. To avoid the phenomenon, IPCTC changes the value of t according to a certain inter-packet delay set. The communication parties share the set.

2.1.2 TRCTC

Cabuk *et al.* [1] later designed a more advanced covert timing channel - TRCTC, which generates a covert channel by replaying the inter-packet delays of legitimate traffic. TRCTC collects the legitimate traffic as the existing data, and obtains a set S_{in} by sorting the inter-packet delays of legitimate traffic. By finding the intermediate value t_c from S_{in}, S_{in} is divided into two subsets S_0 and S_1. TRCTC encodes a 1-bit by randomly replaying a inter-packet delay t_x from S_0 and encodes a 0-bit by randomly replaying a inter-packet delay T_y from S_1. The communication parties share t_c, so the receiver recovers the covert information by receiving the inter-packet delays and t_c. The reason why it is difficult to detect TRCTC is that its statistical characteristics are close to that of the legitimate traffic.

2.1.3 DMCTC

Liu *et al.* [11] presented a covert timing channel with DMCTC, which is designed to counter the statistical-based detection approaches. DMCTC collects N inter-packet delays of legitimate traffic, and the N inter-packet delays are recorded in L intervals. The boundary block α is found so that the number of inter-packet delays on the left side and on the right side of α are as equal. DMCTC encodes a 1-bit by randomly replaying a inter-packet delay from the left interval of α and encodes a 0-bit by randomly replaying a inter-packet delay from the right interval of α. The receiver calculates the boundary block α of L intervals by the same way, and recovers the covert information according to the N inter-packet delays. DMCTC is similar to TRCTC in that they utilize the legitimate traffic to construct a covert channel. Therefor, the reason why it also is difficult to detect DMCTC is that its distribution characteristic are close to that of the legitimate traffic.

2.1.4 JitterBug

Shah *et al.* [21] designed a covert timing channel called JitterBug. The legitimate package is performed a short delay to construct JitterBug and the communication parties of JitterBug share a value w. JitterBug encodes a 1-bit by increasing an packet delay to a value modulo w and encodes a 0-bit by increasing a packet delay to a value modulo $[w/2]$. The receiver recovers the covert information according to the shared value w. For small values of w, the distribution of JitterBug is similar to that of the legitimate traffic.

2.1.5 Others

Model-Base Covert Timing Channel (MBCTC) [8] is a covert timing channels based on distribution fitting methods. MBCTC counters detection by fitting the statistical distribution of the inter-packet delays of legitimate traffic. The on/off channel [27] is proposed from a stand-alone system, which is a classic binary covert timing channel. TCPScript [12] is a passive network covert timing channels established at the TCP layer. The senders confirm the correctness of the hidden information by observing ACK packet of the receiver, and this method is suitable for constructing a multi-ary channels.

2.2 Detection Tests

There are two broad types of detection approaches: special and generic detection approaches. The special detection approaches are only applied to detect a CTCs algorithm, and the limitation of these approaches is large. The generic detection approaches are able to detect different CTCs algorithms. However, due to the high-speed network environment, they are ineffective to capture CTCs.

2.2.1 Special Detection Tests

Cabuk *et al.* [1] proposed a statistical-based detection approach of TRCTC. Hypothesis Test is used to test whether the set of time intervals of TRCTC and legitimate traffic follow the same distribution. The premise is that it is difficult to get a sorted set of TRCTC, and the set needs to satisfy a certain distribution.

Cabuk *et al.* [9] proposed a detection approach based on laws called *"β-Similarity"* for detecting IPCTC. The authors find that the inter-packet delays variance of legitimate traffic always changes. However, the covert channels don't change the variance of the inter-packet delays when the coding method is unchanged. Later Cabuk *et al.* [2] proposed another laws-based approach for detecting IPCTC called *"Compressibility"*. However, it is not suitable for online detection due to the slow true-positive rates.

2.2.2 Generic Detection Tests

The Kolmogorov-Smirnov Test (K-S test) [17] is a statistical-based detection approach. The approach needs to calculate the distance of the empirical distribution functions of test sample and training sample. To judge whether the inter-packet delays of different channels follow the same distribution according to the distance, thereby the author judges whether the test sample is covert timing channels. Although the K-S test is a generic detection method, it's ineffective for capturing CTCs in a complex and varied network environment. Moreover, the approach can not detect TRCTC, JitterBug and DMCTC.

Gianvecchio *et al.* [7] proposed the most effective detection method currently known as entropy detection. Gianvecchio believe the detection of CTCs is mainly divided into two categories: shape-based detection and rule-based detection. The shape of the inter-packet delays can be described by first-order statistics such as mean, standard deviation and distribution. The regularity of the inter-packet delays can be described by high-order statistics, such as the correlation between data. Therefore, Gianvecchio uses the information entropy and the corrected conditional entropy to describe the shape and regularity of the inter-packet delays. Although two entropy methods could effectively detect most of CTCs, unfortunately they did not detect latest DMCTC.

Shrestha *et al.* [22] proposed a support vector machine framework (SVM test) for reliable detection of covert timing channels. The authors extracted multiple fingerprints (*e.g.* K-S test [17], Entropy test and Corrected Conditional Entropy [7] scores) from the network traffic and used them as features to train the support vector machine. Through the classifier of the SVM after training to distinguish whether the traffic is overt or covert. Although the SVM test could detect most of covert timing channels, it needs a lot of features and time to train the classifier of framework. In the high-speed network environment, the features of covert timing channels would change due to the influence (*e.g.* network jitter, packet loss) of the network environment, and the classifier needs to make corresponding changes. This means that the SVM test would lack certain detection robustness in the high-speed network environment, that is, it is easy to be affected by the network environment.

3 Perceptual Hash Measures

Perceptual hash [6, 15, 23], unlike traditional hash, is also known as robust hash and digital fingerprinting. It is a one-way mapping from the digital representation of multimedia information to the perceptual digest. The robustness, discrimination and reliability of perceptual hash are of great importance in the field of information security and communication. Over the last decades, perceptual hash was originally applied in image recognition and authentication [10, 19, 24], and later applied in multimedia information such as audio and video [14, 20, 26]. However, few of perceptual hash technique is applied to the detection of CTCs in the network system. In this sec-

tion, the detail description of the perceptual hash-based approach designed and the approach used to accomplish the detection of covert traffic are provided.

3.1 System Model

Figure 3 shows the model representation of the detection framework. The model is a essential network monitor that has access to the network traffic which is attempting to detect. It could be designed to simply tap into all legitimate network traffic as shown in figure. The model consists of four primary units-a traffic filter, a perceptual feature extractor, the perceptual hash functions and a perceptual hash matching detector. The traffic filter selects network traffic for the perceptual feature extraction. The perceptual feature extractor derives the perceptual features from the traffic selected by the traffic filter. The extracted features are then transformed into the perceptual hash sequences by the perceptual hash functions. By performing the detection threshold estimation, it is ready for implementation on a high-speed network for detection of CTCs.

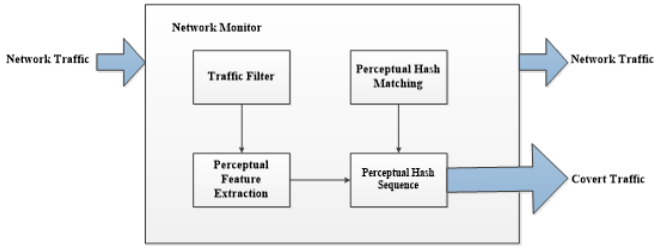

Figure 3: A perceptual hash-based system model for detecting CTCs

3.2 The System Model-based Perceptual Hash Design Process

In this section, we demonstrate the way of designing each unit of the system model. The specific content is as follows:

The perceptual features extraction: First, the network traffic of CTCs is converted into the time series signal $A(t)$, t is the serial number of inter-packet delay. We then extract the perceptual features of $A(t)$ in time-domain and in frequency-domain. The extracted features in frequency-domain are defined as matrix T and the extracted features in time-domain are defined as matrix T_1.

The perceptual hash functions design: H_i=$PH_{gen}(T)$, where PH_{gen} represents a perceptual hash generation function and H_i represents a perceptual hash sequence generated by the matrix T and T_1.

The perceptual matching algorithm design: PD=$PH_{match}(H_i, H_j)$, where PH_{match} is the matching function, H_i and H_j represent the perceptual hash sequences of unknown traffic and sample traffic. PD is the perceptual distance between two sequences that is used to identify the perceptual hash value. It is determined whether H_i and H_j are the same channels by evaluating whether PD is within the detection threshold (β) estimated by the sample traffic. The design process of perceptual hash-approach is shown in Figure 4.

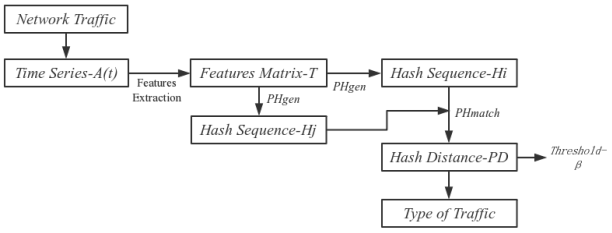

Figure 4: The design of perceptual hash process

3.2.1 The Perceptual Features Extraction Process

Since the law of CTCs in time-domain and the features in frequency-domain are different from the legitimate traffic, we use the discrete wavelet transform (DWT) [18] and the information entropy [4, 13] to extract the perceptual features in frequency-domain and in time-domain, respectively. The extraction process is as follows:

Step 1. DWT analysis: The time series $A(t)$ of CTCs is performed to the global DWT to obtain the high frequency coefficients Hg={Hg_i | i=1, 2, ..., n} and the low frequency coefficients Lh={Lh_j | j=1, 2, ..., m}. Where n and m are the length of high frequency and low frequency coefficients, respectively.

Step 2. Blocking the coefficients: Hg and Lh are divided into non-overlapping range blocks of fixed size. The block length of Hg and Lh are N and M, respectively. The block number of Hg and Lh are S. The obtained matrix T is shown in Equation (1):

$$T = \begin{bmatrix} Lh^1 & Lh^2 & \cdots & Lh^M \\ Lh^{M+1} & Lh^{M+2} & \cdots & Lh^{2\times M} \\ \vdots & \vdots & \vdots & \vdots \\ Lh^{(S-1)\times M+1} & Lh^{(S-1)\times M+2} & \cdots & Lh^{S\times M} \\ Hg^1 & Hg^2 & \cdots & Hg^N \\ Hg^{N+1} & Hg^{N+2} & \cdots & Hg^{2\times N} \\ \vdots & \vdots & \vdots & \vdots \\ Hg^{(S-1)\times N+1} & Hg^{(S-1)\times N+2} & \cdots & Hg^{S\times N} \end{bmatrix} \quad (1)$$

Step 3. The perceptual features extraction in frequency-domain: The standard deviation of each column of the matrix T is calculated, as shown in Equation (2), where v is the mean of each column value of T. The

features parameter vector H_1={$std(k)$ | k=1, 2, ..., N+M} is generated by compressing T.

$$H_1 = \begin{bmatrix} std(1) \\ std(2) \\ \vdots \\ std(N+M) \end{bmatrix}$$

$$std(k) = \sqrt{\frac{1}{S}\sum_{m=1}^{S}(T(m,k)-v)^2} \quad (2)$$

Step 4. The corrective perceptual features extraction: The short-term energy is used to compensate for some frequency domain features due to segmentation loss. The energy value of each row of T is calculated as show in Equation (3). The calculation result is generated as the corrective features parameter vector H_2={$g(k)$ | k=1, 2, ..., S}.

$$H_2 = \begin{bmatrix} g(1) \\ g(2) \\ \vdots \\ g(S) \end{bmatrix}$$

$$g(k) = 10\log\sum_{m=1}^{N+M} T(k,m). \quad (3)$$

Step 5. The perceptual features extraction in time-domain: In order to extract the features in time-domain, the time series $A(t)$ is reasonably segmented. The segment length is C and the number of segments is D. The obtained matrix T_1 is shown in Equation (4).

$$T_1 = \begin{bmatrix} X_1^1 & X_1^2 & \cdots & X_1^C \\ X_2^1 & X_2^2 & \cdots & X_2^C \\ \vdots & \vdots & \vdots & \vdots \\ X_D^1 & X_D^2 & \cdots & X_D^C \end{bmatrix} \quad (4)$$

We calculate the information entropy value G of each row of matrix T_1 and obtain the perceptual features in time-domain. The obtained features parameter vector H_3 is shown in Equation (5).

$$H_3 = \begin{bmatrix} G_1 & G_2 & \cdots & G_D \end{bmatrix} \quad (5)$$

3.2.2 The Perceptual Hash Functions Design

Through the features extraction, we obtain three parameter vectors H_1, H_2 and H_3 respectively. The binary perceptual hash construction is performed on H_1 and H_3, the designed perceptual hash function in Equation (6) is shown. A perceptual sequence $ph_1(k)$={$ph_1(k)$ | k=1, 2, ..., N+M} in frequency-domain and a sequence $ph_3(k)$={$ph_3(k)$ | k=1, 2, ..., D} in time-domain are generated. The three binary hash construction is performed on H_2, The designed perceptual hash function is shown in Equation (7). A perceptual sequence $ph_2(k)$={$ph_2(k)$ | k=2, 3, ..., S-1} corresponding to the corrective features is generated.

$$b_i = \begin{cases} 0 & \text{if } H(k) \le H_{Mean} \\ 1 & \text{otherwise} \end{cases} \quad (6)$$

$$ph(k) = \begin{cases} 1 & \text{if } H(k)^2 - H(k-1)\times H(k+1) > 0 \\ 0 & \text{else if } H(k) - H(k-1) > 0 \\ -1 & \text{otherwise} \end{cases} \quad (7)$$

3.2.3 The Perceptual Hash Matching Algorithm Design

In this paper, the uniformly Hamming distance $D(:,:)$ is used as the method of calculating PD, which is the Bit Error Rate (BER). The calculation formula is as shown in Equation (8), where {i | i=*Normal*, *IPCTC*, *TRCTC*, *JitterBug*, *DMCTC*}, A is the time series of unknown traffic.

$$BER = D(ph(A), ph(A_i)) = \frac{\sum_{j=1}^{3N}|ph_A(j) - ph_{A_i}(j)|}{3N} \quad (8)$$

The perceptual hash sequences of unknown traffic, legitimate traffic, IPCTC, TRCTC, DMCTC and JitterBug sample traffic are generated by the perceptual hash functions. The results are represented as ph, ph_N, ph_{IP}, ph_{TR}, ph_J, ph_{DM}, respectively. The BER of ph to ph_N, ph_{IP}, ph_{TR}, ph_J, ph_{DM} is calculated according to Equation (8). The results are expressed as BER_N, BER_{IP}, BER_{TR}, BER_J, BER_{DM}, respectively. The detection threshold is estimated as β by calculating BER between the homogeneous channels, where {β | β_N, β_{IP}, β_{TR}, β_{DM}, β_J}. Thus, the matching and detection process of unknown traffic is as in Algorithm 1.

Algorithm 1 The matching and detection process

1: Begin
2: **if** $BER_N < \beta_N$ and $\max\{BER_{IP}, BER_{TR}, BER_J, BER_{DM}\} < \min\{\beta_{IP}, \beta_{TR}, \beta_{DM}, \beta_J\}$ **then**
3: $ph \in ph_N$.
4: **else if** $BER_{IP} < \beta_{IP}$ and $\max\{BER_N, BER_{TR}, BER_J, BER_{DM}\} < \min\{\beta_N, \beta_{TR}, \beta_{DM}, \beta_J\}$ **then**
5: $ph \in ph_{IP}$.
6: **else if** $BER_{TR} < \beta_{TR}$ and $\max\{BER_{IP}, BER_N, BER_J, BER_{DM}\} < \min\{\beta_{IP}, \beta_N, \beta_{DM}, \beta_J\}$ **then**
7: $ph \in ph_{TR}$
8: **else if** $BER_J < \beta_{DM}$ and $\max\{BER_{IP}, BER_{TR}, BER_N, BER_{DM}\} < \min\{\beta_{IP}, \beta_{TR}, \beta_N, \beta_J\}$ **then**
9: $ph \in ph_J$
10: **else if** $BER_{DM} < \beta_J$ and $\max\{BER_{IP}, BER_{TR}, BER_J, BER_N\} < \min\{\beta_{IP}, \beta_{TR}, \beta_{DM}, \beta_N\}$ **then**
11: $ph \in ph_{DM}$
12: **end if**
13: If the matching result does not meet the above conditions, we could determine that the unknown traffic is unrecognizable.
14: End

4 Experimental Evaluation

In this section, we verity whether the perceptual hash-based approach is valid through a series of experiments. The perceptual hash-based approach is tested against four typical CTCs: IPCTC [3], TRCTC [1], DMCTC [11], JitterBug [21]. Furthermore, we compare the perceptual hash-based approach (PER-H-test) with three detection tests: the information entropy test (EN-test) [7], the corrective conditional entropy test (CCE-test) [7] and the SVM test (SVM-test) [22].

More specifically, we evaluate the discrimination between unknown traffic and legitimate traffic, four CTCs traffic by calculating *BER*, where $\{BER \mid BER_N, BER_{IP}, BER_{TR}, BER_{DM}, BER_J\}$. The similarity between unknown traffic and legitimate traffic, four CTCs traffic is evaluated by calculating 100-*BER*. The evaluation results of similarity and discrimination will be determined weather the unknown traffic is accurately detected by the true-positive rates. The true-positive rates represent the ratio at which covert timing channels are accurately detected.

4.1 Experimental Results

In the following, we show our experimental results in detail. Four typical CTCs are IPCTC, TRCTC, DMCTC and JitterBug. The experiments are organized by detecting the difficulty of covert timing channels.

4.1.1 IPCTC

Our first set of experiments is test on IPCTC. IPCTC is the simplest and easily detectable covert channel among four CTCs, because its perceptual features exhibit abnormality in both time-domain and frequency-domain. The abnormality features in frequency-domain of IPCTC are caused by the encoding way. If the time series of IPCTC is random, then we treat the time series as a series of Bernoulli trials. Therefore, the inter-packet delays of the time series are approximate to the Geometric distribution. The abnormality features in time-domain is due to the lack of obvious correlations between the inter-packet delays of IPCTC. This means, the inter-packet delays are determined by the covert information being encoded, not by the foregoing packet delays.

We conducted 100 times the detection test for samples of unknown traffic, legitimate traffic and four CTCs traffic. The similarity (100-*BER*) estimation results between the perceptual sequence of unknown traffic (IPCTC) and IPCTC, TRCTC, DMCTC, JitterBug and the legitimate sample traffic are shown in Figure 5, where $\{BER \mid BER_N, BER_{IP}, BER_{TR}, BER_{DM}, BER_J\}$. We could obverse that the similarity between the unknown traffic (IPCTC) and the perceptual sequence of IPCTC is much higher than that of other traffic samples. This means that the generated perceptual sequences of IPCTC samples could be used to accurately distinguish the traffic of IPCTC from the legitimate traffic and three other CTCs. In addition, the reason why 100-*BER* of legitimate traffic and three other CTCs traffic dose not change is that the encoding method and regularity of IPCTC are quite different from that of four other channels. The distinguishing matrix for detection IPCTC is shown in Table 1. It is seen that the identification accuracy of IPCTC and the legitimate traffic was 100 percent, respectively, when working with traffic sizes of 2,000 and 500 samples. At the same time, the experimental results on IPCTC verify the good discrimination of the perceptual hash-based approach.

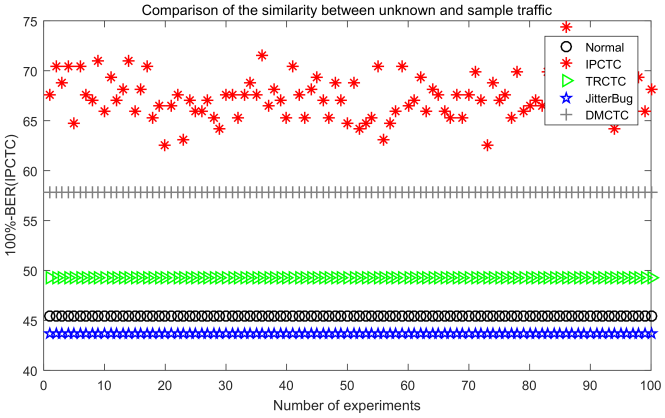

Figure 5: The similarity estimation between IPCTC and other traffic samples

Table 1: The true-positive rates of IPCTC

	Sample Size=2000		Sample Size=500	
	Overt	Covert	Overt	Covert
Overt	100	0	98	2
Covert	0	100	0	100

In order to verify the robustness and practicality of the proposed approach, we run each 100 times test on IPCTC with 10 percent noise (network jitter and 10 percent packet loss) and 30 percent noise (network jitter and 30 percent packet loss) respectively. The robustness estimation results are shown in Figure 6. We could see from Figure 5 that *BER* of DMCTC (45) is closest to IPCTC except IPCTC itself, thus 45 is as the detection threshold β_{IP} of IPCTC. For noiseless and 10 percent noise IPCTC, all *BER* values below 45 mean that the noiseless and 10 percent noise IPCTC can be 100 percent identified. But for the 30 percent noise IPCTC, there is a part of BER value exceeds 45. This means that the 30 percent noise IPCTC could be recognized around 90 percent. In summary, the experimental results verify that the proposed approach for detecting IPCTC has a good detection robustness in a complex network environment.

4.1.2 TRCTC

The second set of experiments is investigated how the proposed approach performs against TRCTC. TRCTC is

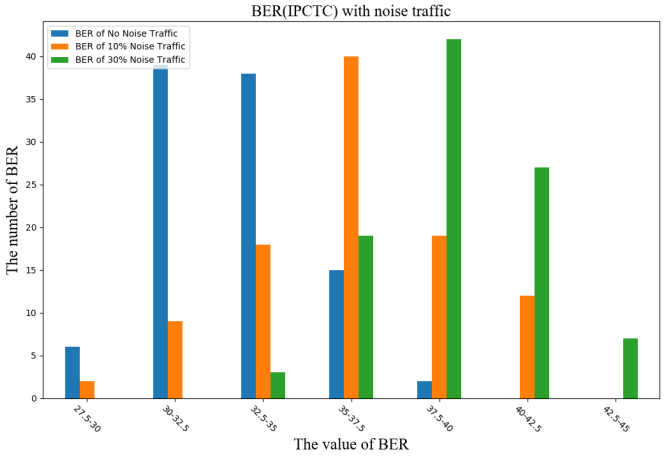

Figure 6: The robustness estimation of the proposed test for detecting IPCTC

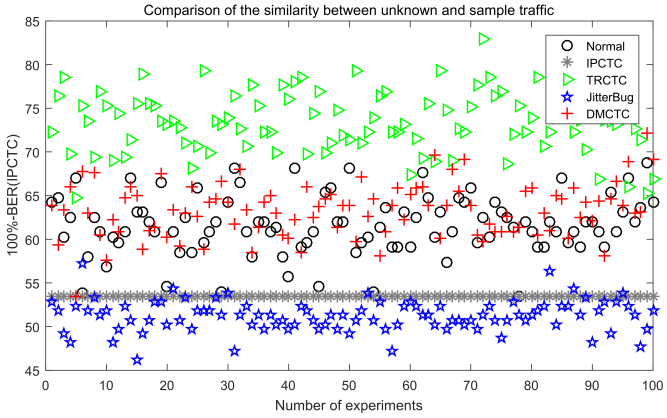

Figure 7: The similarity estimation between TRCTC and other traffic samples

a more advance CTCs, which approximates the behavior of legitimate traffic by replaying a set of legitimate inter-packet delays. Thus, TRCTC has the comparable perceptual features in frequency-domain as legitimate traffic, but the features in time-domain exhibit abnormal. The regularity of TRCTC, like IPCTC, is due to the lack of correlation between inter-packet delays, and the replayed delays still falsifies the regularity of the original process.

We conducted 100 times the detection test for samples of unknown traffic, legitimate traffic and four CTCs traffic. The similarity (100-BER) estimation results between the perceptual sequence of unknown traffic (TRCTC) and IPCTC, TRCTC, DMCTC, JitterBug and the legitimate sample traffic are shown in Figure 7, where $\{BER \mid BER_N, BER_{IP}, BER_{TR}, BER_{DM}\}$. Due to TRCTC traffic is similar to the legitimate traffic, the BER values of some points are below the legitimate traffic samples in the figure. However, the similarity between unknown traffic (TRCTC) and the perceptual sequence of TRCTC is generally higher than that of other traffic samples. We thus could use the generated perceptual sequence of TRCTC samples to distinguish the traffic of TRCTC well from the legitimate traffic and three other CTCs. In addition, the reason why BER of DMCTC traffic is close to the legitimate traffic is that DMCTC has the comparable perceptual features in frequency-domain as legitimate traffic. The distinguishing matrix for detection TRCTC is shown in Table 2. For a traffic size of 500 samples, the proposed approach was able to detect TRCTC with 100 percent. For a traffic size of 2000 samples, the approach was able to detect TRCTC with 81 percent.

We run each 100 times test on TRCTC with 10 percent noise (network jitter and 10 percent packet loss) and 30 percent noise (network jitter and 30 percent packet loss) respectively. The robustness estimation results are shown in Figure 8. We could see from Figure 7 that BER of legitimate traffic (37) is closest to TRCTC except TRCTC itself, thus 37 is as the detection threshold β_{TR} of TRCTC. For the noiseless TRCTC, the values of around 85 percent BER below 37 mean that our approach is able to detect TRCTC with around 85 percent. For the 10 percent and 30 percent noise TRCTC, there are values of around 75 percent BER and around 60 percent BER below 37, respectively. Thus, the proposed approach still has good robustness for detecting TRCTC in the poor network environment.

Table 2: The true-positive rates of TRCTC

	Sample Size=2000		Sample Size=500	
	Overt	Covert	Overt	Covert
Overt	98	2	83	17
Covert	0	100	19	81

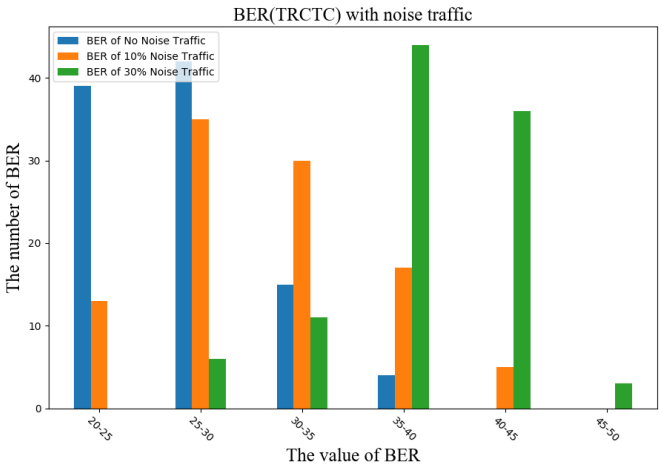

Figure 8: The robustness estimation of the proposed test for detecting TRCTC

4.1.3 DMCTC

Our third set of experiments tested for detecting DMCTC. DMCTC is a more advanced covert timing channel that matches the traffic distribution to imitate the legitimate traffic. By collecting the inter-packet delays of legitimate traffic, DMCTC fits the distribution and replays the inter-packet delays of legitimate traffic to confuse the

detector. Thus, DMCTC has the similar perceptual features to the legitimate traffic in frequency-domain, due to the distribution, and the similar perceptual features in time-domain, due to the packet replay.

We conducted 100 times the detection test for samples of unknown traffic, legitimate traffic and four CTCs traffic. The similarity (100-*BER*) estimation results between the perceptual sequence of unknown traffic (DMCTC) and IPCTC, TRCTC, DMCTC, JitterBug and the legitimate sample traffic are shown in Figure 9, where $\{BER \mid BER_N, BER_{IP}, BER_{TR}, BER_{DM}\}$. We could see it that the similarity between unknown traffic (DMCTC) and the perceptual sequence of DMCTC is generally higher than that of other traffic samples. Since the shape of DMCTC, like TRCTC, simulates the features of legitimate traffic in frequency-domain, the *BER* values of some points are below the legitimate traffic samples in the figure. But the generated perceptual sequence of DMCTC samples could still be used to distinguish the traffic of DMCTC well from the legitimate traffic and three other CTCs. The distinguishing matrix for detection DMCTC is shown in Table 3. For a traffic size of 500 samples, the proposed approach is able to detect DMCTC with 91 percent. For a traffic size of 2000 samples, the approach is able to detect DMCTC with 82 percent.

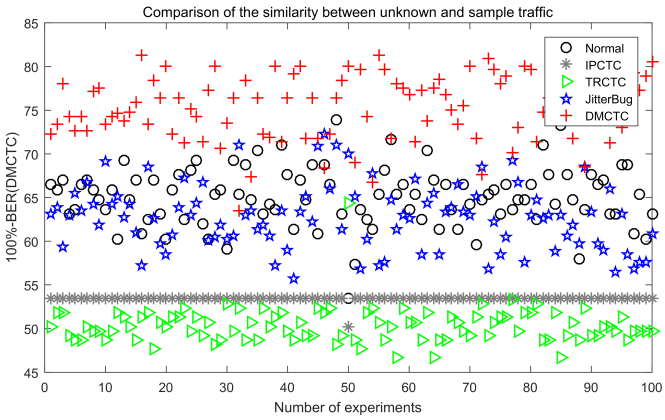

Figure 9: The similarity estimation between DMCTC and other traffic samples

Table 3: The true-positive rates of DMCTC

	Sample Size=2000		Sample Size=500	
	Overt	Covert	Overt	Covert
Overt	79	21	88	12
Covert	18	82	09	91

We run each 100 times test on DMCTC with 10 percent noise (network jitter and 10 percent packet loss) and 30 percent noise (network jitter and 30 percent packet loss) respectively. The robustness estimation results are shown in Figure 10. We could see from Figure 9 that *BER* of legitimate traffic (28) is closest to DMCTC except DMCTC itself, thus 28 is as the detection threshold β_{DM} of DMCTC. For the noiseless DMCTC, the values of 91 percent *BER* below 28 mean that our approach is able to detect DMCTC with 91 percent. For the 10 percent and 30 percent noise DMCTC, there are values of around 70 percent *BER* and 50 percent *BER* below 37, respectively. Since DMCTC is extremely difficult to detect, in the 30 percent noise of cases, there is still a true-positive rate of around 50 percent. Thus, the proposed approach has good detection robustness for detecting DMCTC in the poor network environment.

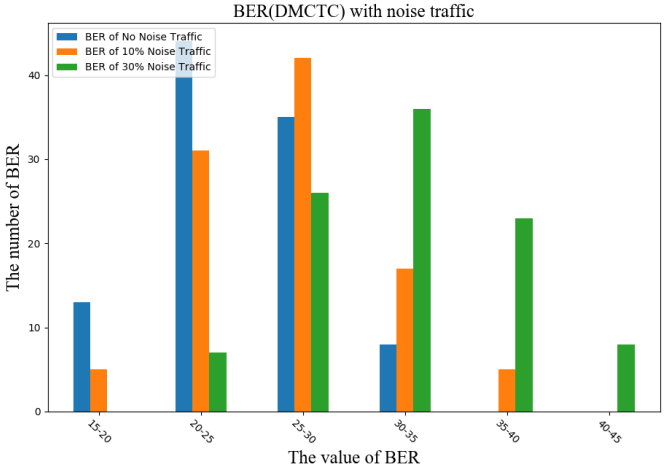

Figure 10: The robustness estimation of the proposed test for detecting DMCTC

4.1.4 JitterBug

The fourth set of experiments is investigated how the proposed approach performs against JitterBug. JitterBug is a passive CTCs, which dose not generate an additional traffic to transmit the covert information. Thus, due to having legitimate traffic as the base and only slightly adding the packet delay, JitterBug preserves partial correlation of the original process. Therefore, JitterBug has similar features in time-domain and in frequency-domain to legitimate traffic. Based on the above reasons, JitterBug is very difficult to detect. Considering that adding the network jitter may affect the channel capacity, we choose the value of w as 15ms to construct JitterBug.

We conducted 100 times the detection test for samples of unknown traffic, legitimate traffic and four CTCs traffic. The similarity (100-*BER*) estimation results between the perceptual sequence of unknown traffic (JitterBug) and IPCTC, TRCTC, DMCTC, JitterBug and the legitimate sample traffic are shown in Figure 11, where $\{BER \mid BER_N, BER_{IP}, BER_{TR}, BER_{DM}\}$. We could see it that the similarity between unknown traffic (JitterBug) and the perceptual sequence of JitterBug is generally higher than that of other traffic samples. Since the shape and regularity of JitterBug perceptual features, is based on the features of legitimate traffic in frequency-domain and time-domain, the *BER* values of some points are below the legitimate traffic samples in the figure. However, the generated perceptual sequence of JitterBug samples could

be used to distinguish the traffic of JitterBug well from the legitimate traffic and three other CTCs. The distinguishing matrix for detection JitterBug is shown in Table 4. For a traffic size of 500 samples, the proposed approach is able to detect JitterBug with 83 percent. For a traffic size of 2000 samples, the approach is able to detect JitterBug with 100 percent.

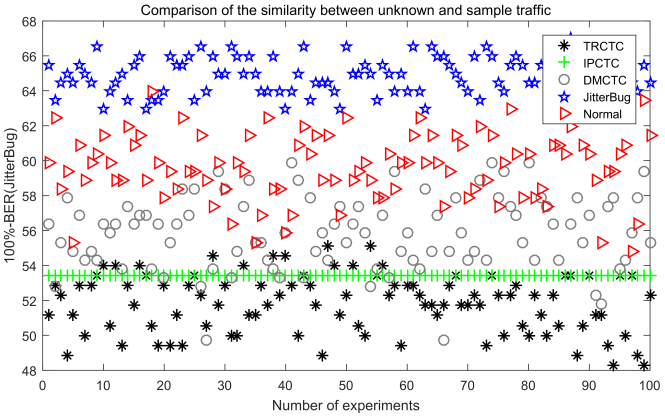

Figure 11: The similarity estimation between JitterBug and other traffic samples

Table 4: The true-positive rates of JitterBug

	Sample Size=2000		Sample Size=500	
	Overt	Covert	Overt	Covert
Overt	97	3	80	20
Covert	0	100	17	83

The 100 times test is performed on detecting JitterBug with 10 percent noise (network jitter and 10 percent packet loss) and 30 percent noise (network jitter and 30 percent packet loss) respectively. The robustness estimation results are shown in Figure 12. We could see from Figure 11 that the *BER* value of legitimate traffic (39) is closest to JitterBug except JitterBug itself, thus 39 is as the detection threshold β_J of JitterBug. For the noiseless JitterBug, the values of around 90 percent *BER* below 39 mean that our approach is able to detect JitterBug with around 90 percent. For the 10 percent and 30 percent noise JitterBug, there are values of around 75 percent *BER* and around 65 percent *BER* below 39, respectively. Since JitterBug is extremely difficult to detect, in 30 percent noise of cases, there is still a detection rate of around 65 percent. Thus, the proposed approach has good robustness for detecting JitterBug in the poor network environment.

4.2 Four Covert Timing Channels Traffic-Variable Sample Size

The last set of experiments is performed to investigate how the perceptual hash-based approach detects with different sample sizes against all four CTCs IPCTC,

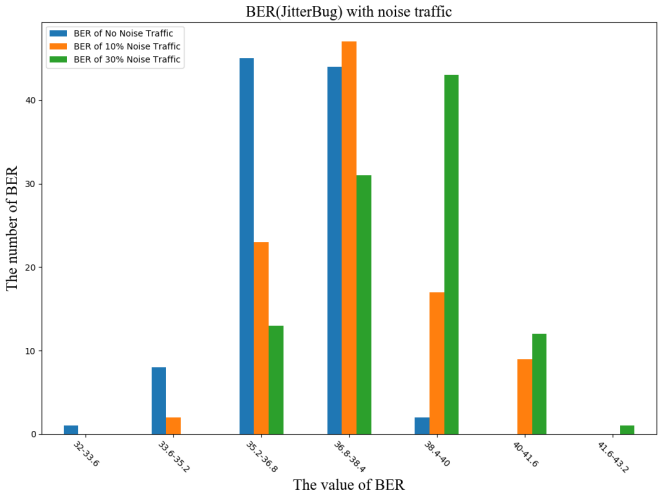

Figure 12: The robustness estimation of the proposed test for detecting JitterBug

TRCTC, DMCTC and JitterBug. The sample size is essential because it is determined the amount of time the detection tests take to detect a covert timing channel. This means that we need to detect CTCs before they transmit as little covert information as possible. Of course, if we detect CTCs with the smallest possible sample size, they will transmit less covert information prior to detection. In general, although the smaller sample size means faster detection, it is often less accurate than the larger sample size. Therefore, we need a compromise between a small sample size and the accuracy of the detection. In this section, we vary sample sizes from 500 to 2250 inter-packet delays for the perceptual hash-based approach (PER-H test), the information entropy approach (EN test) [7], the corrected conditional entropy approach (CCE test) [7] and the support vector machine approach (SVM test) [22].

The true-positive rates for PER-H test, EN test, CCE test and SVM test against IPCTC, TRCTC, DMCTC and Jitterbug with 500 to 2250 inter-packet delays are shown in Figure 13. For these four detection tests, the true-positive rates decreases with the decrease of sample size in different covert timing channels. For IPCTC (Figure 13(a)), there is no decrease in true-positive rates expect CCE test. The regularity of IPCTC is obvious, thus the detection tests doesn't need a large amount of data to detect IPCTC expect CCE test. For TRCTC (Figure 13(b)), En test is unable to detect TRCTC, thus its true-positive rate remain close to 0. In addition, the true-positive rates of the other three approaches all decrease at different rates with the decrease of the sample size, and the declining trend of our approach (PER-H test) is more gentle. Therefore, in the case of small sample size, PER-H test is superior to the other three approaches on detecting TRCTC. For DMCTC (Figure 13(c)), the DMCTC true-positive rates of the approaches mentioned in this section except PER-H test degrade at different rates with the decrease of sample. The DMCTC true-positive rates of PER-H test elevate with the decrease of sample size. Thus, on detecting DMCTC, PER-H test is superior

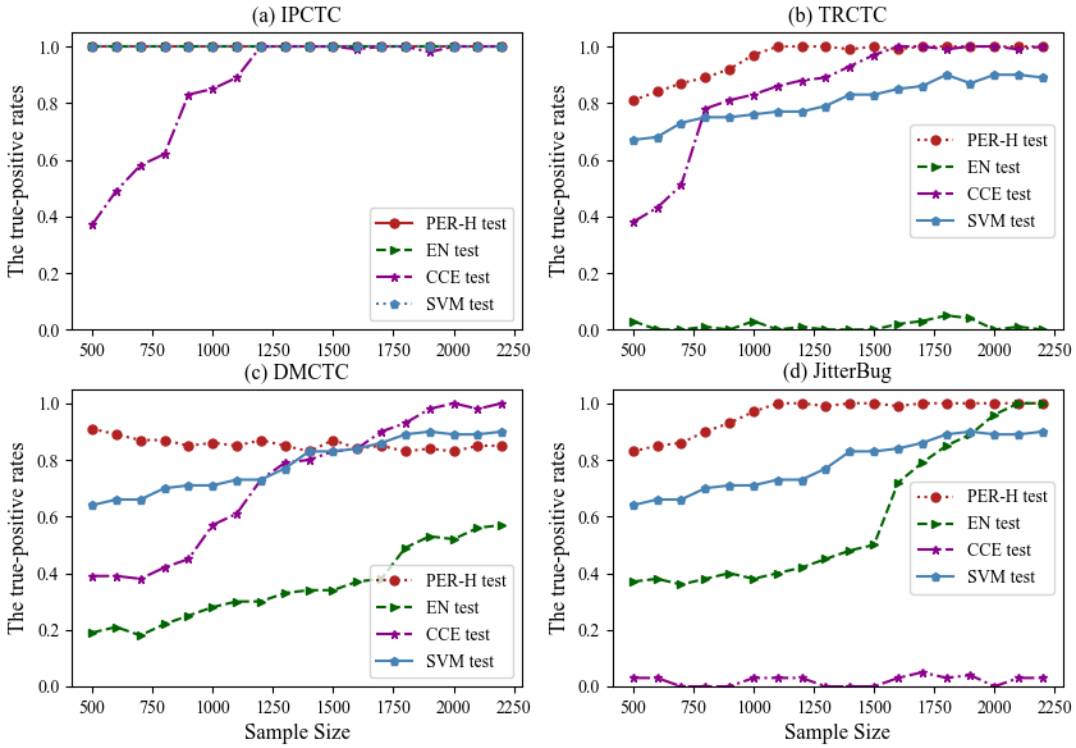

Figure 13: The true-positive rates of all channels-variable sample size

to three other approaches in the case of small sample size and is inferior to three other approaches in the case of large sample size.

Compare with IPCTC, TRCTC and DMCTC, JitterBug (Figure 13(d)) is relatively difficult to detect. It adds only small delays and does not affect the channel capacity. Therefore, it is difficult to distinguish JitterBug from the inter-packet delays of legitimate. CCE test is unable to detect JitterBug, thus its true-positive rate remain close to 0. In the case of small sample size, the true-positive rates of EN test and SVM test is lower than PER-H test. This is because the proposed approach combines the perceptual features of time-domain and frequency-domain and uses the analysis of perceptual hash to find out the weak discrimination between JitterBug and the legitimate traffic. Therefore, PER-H test is superior to EN test and SVM test on detecting JitterBug in both the case of small and large sample size.

In general, we could observe that IPCTC and TRCTC are easier to be detected than DMCTC and JitterBug. The approach in this paper is able to accurately detect IPCTC, TRCTC and JitterBug at the true-positive rates of 1.0 with a small sample size. DMCTC is much more difficult to detect than three other covert timing channels, the season is attributed to the fact that DMCTC not only imitates the regularity and shape of legitimate traffic, but also has a striking similarity with TRCTC and JitterBug. Although the proposed approach is unable to detect DMCTC at the true-positive rate of 1.0, it is superior to EN test, CCE test and SVM test in the case of small sample size.

4.3 Discussion

Our approach is able to detect multiple covert timing channels under certain conditions. For the previous detection approaches, they are unable to detect most of the tested covert timing channels in the high-speed network environment. The main reason is that the detection approaches lose the robustness of detection. For example, the information entropy and corrected conditional entropy approach are incapable of recognizing most covert timing channels in a poor network environment (including network packet loss, jitter, packet injection, *etc.*).

Another reason is that the detection methods need to adapt to the high-speed network environment. This means that the detection methods need detect them under the premise that covert timing channels transmit as little covert information as possible. However, the previous detection approaches are not ideal in the case of small sample size, like the approaches mentioned in Section 4.2.

Our method is proved to be more efficient than the previous detection methods. The proposed approach uses the robustness of perceptual hash to solve the problem that the detection robustness is loss in the case of network interference. At the same time, our method combines the perceptual features in time-domain and in frequency-domain, and solves the problem that the true-positive rates is low in the case of small sample size by the discrimination of the perceptual hash. Thus, the perceptual hash-based approach has effectively passed the test of detection covert timing channels.

5 Conclusion and Future Works

Existing detection methods could detect most of covert timing channels, while lack certain detection robustness. At the same time, the true-positive rates of existing methods are not ideal in the case of small sample size. For the above shortcomings of methods, we proposed a perceptual hash-based approach to detect covert timing channels. The proposed approach improves the discrimination between CTCs traffic and the legitimate traffic by combining the perceptual features in time-domain and in frequency-domain of CTCs. This makes the true-positive rates of proposed approach on detecting CTCs is superior to others in the case of small sample size. In the meantime, the proposed approach utilizes the robustness of perceptual hash to preserve the main of features. This means that our approach could still identify covert timing channels well in the case of network interference (network jitter, data packet loss *e.g.*). Experimental results confirm that the proposed approach not only has a good true-positive rate in the case of small sample size, but also has good detection robustness in the case of network interference.

In future, we plan to optimize the extracted perceptual features and improve the designed perceptual hash function. In addition, we plan to utilize the perceptual hash-based approach against the new covert timing channels in the future.

Acknowledgments

This study was supported by the National Natural Science Foundation of China (NO.61370007), the Program for New Century Excellent Talents of Fujian Provincial (NO.2014FJ-NCET-ZR06), and the Subsidized Project for Postgraduates' Innovative Fund in Scientific Research of Huaqiao University (No.17014083018).

References

[1] S. Cabuk, Network Covert Channels: Design, Analysis, Detection, and Elimination, 2006. (`https://core.ac.uk/download/pdf/21173179.pdf`)

[2] S. Cabuk, C. E. Brodley, and C. Shields, "Ip covert channel detection," *Acm Transactions on Information & System Security*, vol. 12, no. 4, pp. 1–29, 2009.

[3] S. Cabuk, C. E. Brodley, and C. Shields, "IP covert timing channels: Design and detection," in *Acm Conference on Computer & Communications Security*, pp. 178–187, 2004.

[4] Y. Chen, N. Zhang, H. Tian, T. Wang, and Y. Cai, "A novel connection correlation scheme based on threshold secret sharing," *IEEE Communications Letters*, vol. 20, no. 12, pp. 2414–2417, 2016.

[5] Y. Chen, X. Ma, and X. Wu, "DDoS detection algorithm based on preprocessing network traffic predicted method and chaos theory," *IEEE Communications Letters*, vol. 17, no. 5, pp. 1052–1054, 2013.

[6] L. Chen, Z. Li, and J. F. Yang, "Compressive perceptual hashing tracking," *Neurocomputing*, vol. 239, pp. 69–80, 2017.

[7] S. Gianvecchio and H. Wang, "An entropy-based approach to detecting covert timing channels," *IEEE Transactions on Dependable & Secure Computing*, vol. 8, no. 6, pp. 785–797, 2011.

[8] S. Gianvecchio, H. Wang, D. Wijesekera, and S. Jajodia, "Model-based covert timing channels: Automated modeling and evasion," in *International Symposium on Recent Advances in Intrusion Detection*, pp. 211–230, 2008.

[9] C. G. Girling, "Covert channels in Lan's," *IEEE Transactions on Software Engineering*, vol. SE-13, no. 2, pp. 292–296, 1987.

[10] J. Ji, X. Lü, L. Han, and C. Zhang, "Fast and adaptive region merging based on perceptual hashing via multi-thresholding for SAR image segmentation," *Remote Sensing Letters*, vol. 7, no. 12, pp. 1199–1208, 2016.

[11] G. Liu, J. Zhai, and Y. Dai, "Network covert timing channel with distribution matching," *Telecommunication Systems*, vol. 49, no. 2, pp. 199–205, 2012.

[12] X. Luo, E. W. W Chan, and R. K. C Chang, "TCP covert timing channels: Design and detection," in *IEEE International Conference on Dependable Systems & Networks with Ftcs & Dcc*, 2008. DOI: 10.1109/DSN.2008.4630112.

[13] X. Ma and Y. Chen, "DDoS detection method based on chaos analysis of network traffic entropy," *IEEE Communications Letters*, vol. 18, no. 1, pp. 114–117, 2014.

[14] R. D. Major, *Pre-Distribution Identification of Broadcast Television Content using Audio Fingerprints*, US20180359540A1, 2014. (`https://patents.google.com/patent/US20180359540A1/en`)

[15] A. Neelima and K. M. Singh, "Perceptual hash function based on scale-invariant feature transform and singular value decomposition," *Computer Journal*, vol. 59, no. 9, pp. 1275–1281, 2016.

[16] E. U. Opara and O. A. Soluade, "Straddling the next cyber frontier: The empirical analysis on network security, exploits, and vulnerabilities," *International Journal of Electronics & Information Engineering*, vol. 3, no. 1, pp. 10–18, 2015.

[17] P. Peng, P. Ning, and D. S. Reeves, "On the secrecy of timing-based active watermarking trace-back techniques," in *IEEE Symposium on Security & Privacy*, 2006. (`http://discovery.csc.ncsu.edu/pubs/Oakland06.pdf`)

[18] J. Qin, R. Sun, X. Xiang, H. Li, and H. Huang, "Anti-fake digital watermarking algorithm based on QR codes and DWT," *International Journal of Network Security*, vol. 18, no. 6, pp. 1102–1108, 2016.

[19] S. B. Qiao, Q. Y. Zhang, T. Zhang and D. F. Wu, "Spectrogram-based efficient perceptual hashing scheme for speech identification," *International Journal of Network Security*, vol. 21, no. 2, pp. 259–268, 2019.

[20] N. Saikia, and P. K. Bora, "Perceptual hash function for scalable video," *International Journal of Information Security*, vol. 13, no. 1, pp. 81–93, 2014.

[21] G. Shah, A. Molina, and M. Blaze, "Keyboards and covert channels," in *Conference on Usenix Security Symposium*, vol. 15, no. 5, 2006.

[22] P. Shrestha, M. Hempel, F. Rezaei, and H. Sharif, "A support vector machine-based framework for detection of covert timing channels," *IEEE Transactions on Dependable & Secure Computing*, vol. 13, no. 2, pp. 1–1, 2016.

[23] X. Wang, K. Pang, X. Zhou, Z. Yang, L. Lu, and J. Xue, "A visual model-based perceptual image hash for content authentication" *IEEE Transactions on Information Forensics & Security*, vol. 10, no. 7, pp. 1336–1349, 2015.

[24] F. Wen, H. M. Hu, Z. Hu, S. Liao, and L. Bo, "Perceptual hash-based feature description for person re-identification," *Neurocomputing*, vol. 272, pp. 520–531, 2017.

[25] X. Wu and Y. Chen, "Validation of chaos hypothesis in nada and improved ddos detection algorithm," *Communications Letters IEEE*, vol. 17, no. 12, pp. 2396–2399, 2013.

[26] G. Yang, X. Chen, and D. Yang, "Efficient music identification by utilizing space-saving audio fingerprinting system," in *IEEE International Conference on Multimedia & Expo*, 2014. https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6890236

[27] S. Zander, G. Armitage, and P. Branch, "Stealthier inter-packet timing covert channels," in *International Ifip Tc 6 Conference on Networking*, pp. 458–470, 2011.

Biography

Linfan Wang was born in Shanxi, China in 1995. He received the B.S. Degree from Hubei University of Mediciney, Hubei, China in 2017. He is currently pursuing the M.S. Degree in Huaqiao University. His research interests include Covert Channels Detection and Perceptual Hash and Blockchain and Application.

Yonghong Chen received the Ph.D. degree from Chongqing University, Chongqing, China, in 2005. He is a Professor in Huaqiao University of China. His current interests include Network and Information Security, Network intrusion detection, Digital Watermarking and Property Protection and Blockchain and Application.