

Robust and Undetectable Covert Timing Channels for i.i.d. Traffic

Abstract. Covert timing channels exploit inter-packet delays in network traffic to transmit secret messages. The two most important design goals are undetectability (the covert channel has to remain hidden to a potential adversary that is monitoring the inter-packet delay pattern) and robustness (messages can be decoded correctly even in presence of (maliciously) injected noise). In previous proposals undetectability has been only validated against a set of known statistical methods, leaving the resistance against possible future attacks unclear. Moreover, many existing schemes do not provide any robustness at all.

In this paper, we introduce a covert timing channel that is both robust and provably undetectable for network traffic with independent and identically distributed (i.i.d.) inter-packet delays. I.i.d. traffic models are very useful because they are simple to analyze, and constitute essential elements of many advanced network traffic models such as Markov Renewal Processes (MRP) and Batch Renewal Processes (BRP) with applications in video streaming and Voice over IP (VOIP). As opposed to recent work we do not rely on any strong assumptions, e.g., bounded jitter, but require only the existence of a cryptographically secure pseudorandom generator. We verify the effectiveness of our approach by conducting a series of experiments on Telnet traffic and discuss the trade off between various encoding and modulation parameters.

1 Introduction

Covert channels aim at establishing a communication channel hidden from any outsider. There is a large body of literature on covert channels in computer networks that exploit open overt communication (such as certain network protocols) as the carrier medium to transmit secret messages [24]. A specific type of covert channels are covert timing channels which exploit timing information and predominantly inter-packet delays to transmit covert messages [1, 4, 17, 23]. Most existing methods to establish covert timing channels have been successfully defeated using statistical tests [1, 4, 23] or entropy-based approaches [9]. Recent research started to incorporate traffic models into the covert timing channel design process to evade potential detection schemes [10, 15, 22].

However, these solutions have deficiencies regarding various aspects such as the required resources (e.g., to update and transmit the system model parameters regularly) [10], the underlying assumptions (e.g., bounded jitter) [22] or the limited adversary model (e.g., offer security only against specific statistical tests) [10, 15]. To the best of our knowledge, none of the existing solutions to covert timing channels provides a scheme that is both *robust* and *provably undetectable*. Here, undetectability means the incapability of the adversary to detect the covert channel by distinguishing between covert and overt communications. Robustness means that the covert messages can be correctly decoded, even in the presence of (possibly maliciously) injected noise.

In this paper we focus on constructing covert timing channels that are *both* robust and provably undetectable. Here, we use the traffic models where inter-packet delays are independent and identically distributed (i.i.d.) random variables, as recently proposed in [22]. Among the set of possible traffic models, i.i.d. traffic model represents a basic model that is extensively used in network analysis [6, 8, 13, 18]. When modeling network traffic, packet arrivals and connection requests are often assumed to follow a Poisson process because of their attractive theoretical properties [13]; the i.i.d. Pareto distribution has also been used to model inter-packet delays of Telnet traffic [18]. Although there are only a limited number of real traffic that strictly follow the i.i.d. distribution, i.i.d. is the essential element in many advanced traffic models and can be readily extended to other more general traffic models with reasonable analytical adjustments. For example, most multimedia applications, such as video steaming [16] and Voice over IP (VOIP) [12], can be well modeled as Markov Renewal Processes, where multiple traffic sources are modeled as a multiple-state Markov

Chain and each traffic source is i.i.d. For more general traffic, recent research has shown that it can be well modeled by Batch Renewal Processes (BRP) [14], which is considered most accurate to quantify the traffic correlation among existing models. Specifically, in a BRP, the number of arrivals in different batches are i.i.d. and the intervals between batches are also i.i.d. Therefore, by partitioning the transmitted data into multiple groups and encoding each group using different i.i.d. traffic sources, the dynamic properties of most popular traffic models can be preserved at the same time.

Contribution and outline. In this paper, we propose a covert timing channel that is *both* provably undetectable and robust for any legitimate traffic whose inter-packet delays are i.i.d., while following an arbitrary distribution. Toward this goal, we first adopt the idea of spreading codes in the encoding process in Section 4 to mitigate the impact of transmission noise. Spreading codes have been successfully applied to construct a robust covert timing channel [15] for generic traffic, but undetectability was only guaranteed with respect to two commonly used statistical tests. Second, in Section 5, the encoding scheme is applied to a more specific traffic model in order to achieve undetectability against *any* (efficiently computable) statistical test; we will call this security property *polynomial undetectability* in the sequel. Finally, through experimental results using different types of noise and noise power levels, we verify the effectiveness of our robust encoding scheme and compare our results to the most recently proposed covert channel [22], which, to our knowledge, is the only covert timing channel that is shown to achieve polynomial undetectability for i.i.d. traffic. Specifically, our scheme improves the robustness of the covert communication and achieves true polynomial undetectability without relying on strong assumptions such as bounded jitters, which is assumed in the existing work. Further, a tradeoff is shown between robustness and attainable transmission rate by adjusting the encoding and modulation parameters. To validate the effectiveness of our approach, we test our covert timing channel using model traffic extracted from Telnet applications whose inter-packet delays are shown to be i.i.d. [18]. The results in Section 6 show that the generated covert traffic can masquerade well as legitimate traffic. At the same time, under various network conditions with delay and jitter, our proposed approach can achieve any robustness requirement at the price of the transmission rate.

2 Related Work

A covert timing channel involves modulating the packet transmission of legitimate traffic so that confidential information (the covert message) is encoded into packet arrival patterns. The simplest form of a covert timing channel is implemented by a binary on-off transmission scheme [4, 17]: in a specific time interval, a packet arrival indicates the bit 1 and an absence indicates the bit 0. This on-off scheme is the first public implementation of an IP covert channel. However, as mentioned in [4], the inter-packet delays of covert traffic pattern are very regular and can be easily differentiated from legitimate traffic. To avoid detection, later work [5] extended the idea by either increasing the length of inter-packet delays or adding significant noise to the channel. Nonetheless, its security is only experimentally verified against regularity tests [1] that check the variance of packet delays.

A more advanced type of covert timing channel encodes the covert message directly in the time, i.e., inter-packet delays of consecutive packets. Particularly, an interval-time-replay scheme is proposed in [3]. It first partitions the empirical range of the inter-packet delay of legitimate traffic into two equal subsets, corresponding to “small-delays” and “large-delays”. Then, a bit-1 of the covert message is sent by randomly replaying a large delay and a bit-0 by using a small delay. The keyboard JitterBug [23] utilizes the inter-packet delays of an existing interactive session

to reveal secret message. By adding small delays onto the time sequences between key-presses, covert message can be transmitted. Compared to the above on-off schemes, the design of these covert timing channels takes into account some characteristics of the original legitimate traffic. Specifically, the shape of inter-packet delays of legitimate traffic is generally retained by the covert traffic. However, some statistic feature, i.e., the distribution [4], the correlation [1], or the entropy of the traffic [9], is changed by embedding the covert message and such information has been effectively exploited for detection. Recently, a model-based covert timing channel [10] was proposed to thwart these detection methods. The scheme first derives a filter model based on the statistics of the observed legitimate inter-packet delays and uses this model to generate the covert traffic. In order to adaptively fit a non-stationary traffic, however, this scheme requires frequent transmission of the model parameters to the decoder, which results in a considerable system overhead.

A more powerful solution to defend against covert timing channels is to actively disrupt the communication channel. For example, the authors in [11] add random delays to the traffic using a jamming device and show that the throughput of the covert information is vanishing in practice. Although jammers may also reduce the performance of the legitimate traffic at the same time, it introduces another requirement—*robustness*—to covert timing channel designs. Most existing covert timing channels only address the detection challenge; the transmission efficiency in terms of channel robustness has only been considered under limited conditions. In [22], the authors proposed an encoding scheme that achieves near optimal data rate under normal network conditions. Based on this scheme, a timing channel was introduced to generate i.i.d. traffic undetectable for a polynomial time adversary. However, the scheme has limitations due to the strong assumption that the traffic modification introduced during transmission, e.g., the additive jitter, must be bounded within a certain range. Therefore, security is only guaranteed under this condition and the robustness is not sufficient against noisy channels or a malicious jammer. Previous work [15] has shown that spreading codes can effectively increase the robustness against various intentional and/or unintentional channel distortions. In particular, the authors present a method to modulate the (encoded) secret message to mimic the distribution of the inter-packet delays of a non-stationary legitimate traffic. Nonetheless, only two main and commonly used classes of statistical tests, namely shape and regularity tests [1, 4], were applied to validate the undetectability of the scheme. Moreover, the distribution of the covert traffic is only an approximation of the legitimate one and thus it may not provide sufficient security against more sophisticated detection methods developed in the future.

As we will elaborate later, our covert channel design aims to overcome both problems within i.i.d. traffic model, offering undetectability against any polynomial time distinguisher as well as robustness against (malicious) traffic alternation. We will also discuss the trade-off with respect to transmission efficiency and robustness.

3 System Model and Design Criteria

The goal of our work is to design a *robust* and *polynomially undetectable* covert timing channel by manipulating inter-packet delays between successive packets.

3.1 Preliminaries

We will use the terms *covert communication* and *overt communication*, respectively, to refer to a communication with and without embedded covert channel and use *covert traffic* and *legitimate traffic* for the traffic types associated to these two different channels. In the rest of the paper we use the following notation:

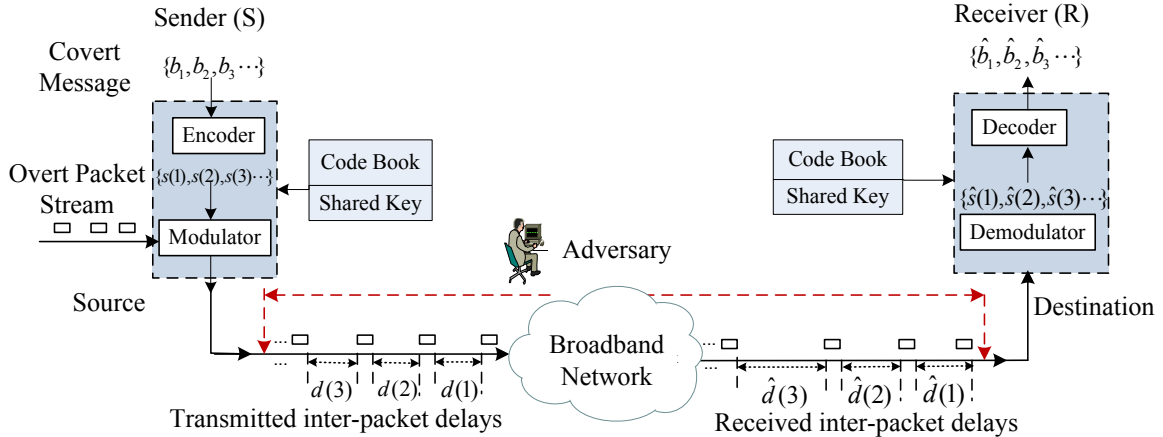


Fig. 1. System level diagram of the proposed covert timing channel.

- b_k : The k -th information bit of the covert message
- $s(n)$: The n -th code symbol encoded with spreading codes
- $d(n)$: The n -th covert inter-packet delay transmitted from the sender
- $\hat{d}(n)$: The n -th covert inter-packet delay received at the receiver
- $\tilde{d}(n)$: The n -th inter-packet delay from legitimate traffic
- N : Spreading factor in the encoding process
- K : Number of orthogonal channels in the encoding process
- M : Number of possible code symbols after the encoding process
- $R_t = K/N$: Transmission rate in bits per packet (bpp)
- P_e : Bit error rate (BER)

Note that we use a parenthesized letter (e.g., $d(n)$) to index the code symbol and packet interval and a subscript (e.g., b_k) to index the information bit.

We also introduce some basic definitions that we use later. A function $f : \mathbb{N} \rightarrow \mathbb{R}$ is called *negligible* if for any $j' \in \mathbb{N}$ exists a polynomial $p(z)$ over \mathbb{N} such that $|f(j)| < 1/p(j)$ for all $j \geq j'$. For a probability distribution \mathbb{D} , the expression $z \leftarrow \mathbb{D}$ denotes the event that z has been sampled according to \mathbb{D} . A *distinguisher* \mathcal{D} is a (possibly probabilistic) algorithm that aims for distinguishing between two different distributions \mathbb{D} and \mathbb{D}' . More precisely, \mathcal{D} receives some values that are either sampled according to \mathbb{D} or \mathbb{D}' and outputs a bit $b \in \{0, 1\}$. The *advantage* of \mathcal{D} is defined by

$$\text{Adv}_{\mathcal{D}}(\mathbb{D}, \mathbb{D}') := |\Pr[\mathcal{D}(z) = 1 | z \leftarrow \mathbb{D}] - \Pr[\mathcal{D}(z) = 1 | z \leftarrow \mathbb{D}']|. \quad (1)$$

Definition 1 (CSPRNG). Let R denote a range of values and G a pseudorandom number generator that produces a sequence of values in R depending on a (secret) seed $s \in \{0, 1\}^\sigma$. For a positive integer $N \geq 1$, we consider two distributions on R^N : \mathbb{U}_N , the uniform distribution, and \mathbb{G}_N , the distribution of N outputs by G . The generator G is called a cryptographically secure pseudorandom generator (CSPRNG) with respect to the size σ of the seed if it holds for any distinguisher \mathcal{D} with a runtime polynomial in σ and N polynomial in σ that $\text{Adv}_{\mathcal{D}}(\mathbb{U}_N, \mathbb{G}_N)$ is negligible in σ .

3.2 System Model

The overall system model is shown in Figure 1. We define the *sender* (S) and the *receiver* (R) entities as the two ends of a covert communication. S has access to some sensitive information

(covert message) to be transmitted to R. To achieve this, S embeds the covert information into a legitimate packet stream. We consider a binary channel, where the covert message is coded as a binary sequence over the alphabet $\{-1, +1\}$. The covert message is $\{b_1, b_2, b_3, \dots\}$, where b_i is the i -th *information bit*. The information bits are encoded to produce *code symbols* $\{s(1), s(2), s(3), \dots\}$, which are finally modulated in the inter-packet delays $\{d(1), d(2), d(3), \dots\}$ of a packet stream that is sent from the source to the destination. To decode the covert message correctly, a code book and a secret key are assumed to be shared between S and R prior to the covert transmission.

Adversary Model. An adversary, e.g., a timing channel jammer or an intrusion detection system, can monitor or manipulate the transmission between S and R. We assume that an adversary has access to both legitimate and covert traffic (but not simultaneously¹), and can easily derive some characteristics (such as the distribution of the inter-packet delays). The adversary also has knowledge of the structure and the modulation algorithm of the covert timing channel. However, he is not able to access the code book and the secret key.

Observe that in our system model, no backward channel exists from R to S. Thus, the embedding of the secret message is independent of any disruption in the communication. In particular, regarding undetectability (see Definition 2 below) it is sufficient to restrict analysis to an attacker that passively listens to the network communication. Nonetheless, the covert channel needs to be robust against network noise which is injected by the network or by an adversary trying to disrupt the covert channel.

3.3 Design Criteria

The design of the covert timing channel is guided by the following two goals:

Undetectability. On a high level, undetectability means that no efficient algorithm can distinguish between inter-packet delays of legitimate traffic and covert traffic. We define a sequence of N inter-packet delays which stems from covert traffic as $\mathbf{d} = (d(1), d(2), \dots, d(N))$ while the ones from legitimate traffic as $\tilde{\mathbf{d}} = (\tilde{d}(1), \tilde{d}(2), \dots, \tilde{d}(N))$. We denote their corresponding distributions by \mathbb{D}_N and $\tilde{\mathbb{D}}_N$, respectively. Next we define our notion of undetectability.

Definition 2 (Polynomial Undetectability). *A covert timing channel is called polynomially undetectable with respect to a security parameter σ if it holds for any distinguisher \mathcal{D} with a runtime polynomial in σ and for any N that is polynomial in σ that $\text{Adv}_{\mathcal{D}}(\mathbb{D}_N, \tilde{\mathbb{D}}_N)$ is negligible in σ .*

For the rest of the paper, we will only consider distinguishers that have a runtime polynomial in a given security parameter σ , and only values N that are polynomial in σ as well.

Robustness. Due to channel unreliability during transmission (routers, firewalls, switches or repeaters can incur processing delays), the inter-packet delays generated at S will be perturbed at R. In addition, a timing jammer can be introduced to reduce the channel capacity of covert communication; this causes further deviation of the delays from their designated values. To resist these unintended and malicious disruption, covert communication must be robust. Particularly, the robustness can be measured as the capability to achieve a decoding bit error rate (BER) $P_e \leq \varepsilon$ under a given robustness requirement $\varepsilon \in \mathbb{R}^+$. P_e inversely proportional to the Signal-to-Noise Ratio (SNR) E_s/E_x [20], where E_s is the signal power and E_x is the noise power. Considering that there is a one-to-one mapping between BER and SNR, we can always achieve any given robustness requirement ε by increasing the SNR. Therefore, robustness can be defined as follows:

¹ Note the covert traffic is generated by modifying legitimate input traffic. Therefore, at any time only a single traffic type (either legitimate or covert) can exist in the transmission path.

Definition 3 (Robustness). A covert timing channel is called γ -gain robust if the SNR after performing the encoding and modulation process is γ times greater than the original SNR, where $\gamma \in \mathbb{R}^+$ (we call γ as robustness gain). When $\gamma \rightarrow +\infty$, the covert timing channel is said to be perfectly robust.

4 Robust Encoding with Spreading Codes

It was shown in [15] that encoding the message using spreading codes can efficiently increase the robustness of covert timing channels. Any arbitrarily strong additive noise can be mitigated by selecting a sufficiently large spreading factor. In this section, we briefly review the general concept of spreading codes and then show how to utilize spreading codes for covert timing channel design.

Using spread encoding, in the simplest case each bit $b_k \in \{+1, -1\}$ of the covert message $\{b_1, b_2, \dots\}$ is encoded into

$$\mathbf{s}_k = b_k \cdot \mathbf{c}, \quad (2)$$

where $\mathbf{c} = (c_1, c_2, \dots, c_N) \in \{\pm 1\}^N$ is an arbitrary code word. By this means, a code word \mathbf{c} of length N will be used to convey just one information bit b_k and the transmission rate R_t after applying the spread encoding process decreases to $1/N$ bit per packet (bpp)². Considering that one of the main advantages of spreading codes is the capability of transmitting different signals simultaneously, we aim at encoding multiple bits at once. Specifically, K code words $\mathbf{c}_1, \dots, \mathbf{c}_K$ can be used to carry K information bits b_1, \dots, b_K over K parallel channels. The symbols from all K channels are combined to a single sequence

$$\mathbf{s} = (s(1), s(2), \dots, s(N)) = \sum_{k=1}^K \mathbf{s}_k = \sum_{k=1}^K b_k \cdot \mathbf{c}_k, \quad (3)$$

with $s(i) \in [-K, K]$, for transmission. To differentiate the transmitted bits for each channel, the spreading codes must be orthogonal to each other, i.e., $\langle \mathbf{c}_i, \mathbf{c}_j \rangle$ equals N if $i = j$ and 0 otherwise. Walsh-Hadamard codes [19] are popular orthogonal codes and will be used for our tests.

Assuming additive noise, the received symbols can be expressed as $\hat{\mathbf{s}} = \mathbf{s} + \mathbf{x}$. Here \mathbf{x} is the process noise after demodulation; \mathbf{s} and $\hat{\mathbf{s}}$ are N -dimensional vectors. To decode the k -th information bit, we apply a threshold rule to the inner product of the received code symbols and the code word \mathbf{c}_k :

$$\hat{b}_k = \frac{1}{N} \langle \hat{\mathbf{s}}, \mathbf{c}_k \rangle = \frac{1}{N} \langle \mathbf{s}, \mathbf{c}_k \rangle + \frac{1}{N} \langle \mathbf{x}, \mathbf{c}_k \rangle = \underbrace{\sum_{i=1}^K \frac{b_i}{N} \langle \mathbf{c}_i, \mathbf{c}_k \rangle}_{=b_k} + \frac{1}{N} \langle \mathbf{x}, \mathbf{c}_k \rangle. \quad (4)$$

If \mathbf{x} and \mathbf{c}_k are uncorrelated (e.g., orthogonal), \hat{b}_k is equal to b_k and can be recovered by choosing a proper threshold (e.g., 0 for a binary sequence).

Since N is the length of the spreading code and the maximum number of orthogonal channels, K must be less or equal to N . Therefore, for a binary channel, the maximum transmission rate R_t we can achieve is 1 bpp in the case of $K = N$.

We assume that the input covert message is composed of random binary bits. This is always achievable, as the covert message can be encrypted under a semantically secure cipher or compressed. In addition, a good spreading code should have the properties of a pseudo random sequence,

² Directly transmitting binary information bits per packet can achieve 1 bpp.

such as balance (i.e., on average the same number of 1's and -1 's) and short run length (low average number of consecutive 1's or -1 's). Therefore, the code symbol in each channel can be regarded as a random binary sequence with equal probability, that is, $Pr[s_k(n) = 1] = Pr[s_k(n) = -1] = 1/2$. For simplicity, we omit the code symbol index n in the following. Let k_1 be the number of channels with the code value $s_k = 1$ and k_2 be the one with the code value $s_k = -1$; k_1 and k_2 are random variables with $0 \leq k_1 \leq K$, $0 \leq k_2 \leq K$ and $k_1, k_2 \in \mathbb{N}$. As there are K channel in total, we have $K = k_1 + k_2$. From Eq. (3), the code symbol s is the sum of encoded symbols at each channel. Then we have $s = k_1 * 1 + k_2 * (-1) = k_1 - k_2$. Since each channel carries an independent bit of the input binary sequence, the encoded symbols s_k at each channel are independent and the probability mass distribution (PMF) of code symbol s is given by

$$P_s(l) = Pr[s = l] = Pr[k_1 - k_2 = l] = Pr[k_2 = \frac{K-l}{2}] = \begin{cases} \binom{K}{\frac{K-l}{2}} (\frac{1}{2})^K & K-l \text{ even} \\ 0 & \text{otherwise,} \end{cases} \quad (5a)$$

where $-K \leq l \leq K$ and $l \in \mathbb{Z}$. Therefore, the distribution of a code symbol s is an up-sampled version³ of the PMF of a binomial distribution $B(K, 1/2)$.

5 Construction

For legitimate traffic whose inter-packet delays $\{\tilde{d}(1), \tilde{d}(2), \dots\}$ are i.i.d. random variables, the inter-packet delays are fully determined by their probability distribution, e.g., the cumulative distribution function (CDF) $F_{\tilde{d}}(\cdot)$. Therefore, the goal of undetectability is to provide an efficient generator of perfectly random inter-packet delays $\{d(1), d(2), \dots\}$ that follow the specific known distribution of the legitimate traffic, but encode a covert message.

5.1 Modulation

Since the transformation of a code symbol $s(n)$ to the inter-packet delay $d(n)$ must be invertible, we consider a one-to-one mapping:

$$d(n) := T(s(n)), \quad n = 1, 2, \dots, \quad (6)$$

where $T(\cdot)$ is an invertible function. The demodulation at the receiver is done by:

$$\hat{s}(n) := T^{-1}(\hat{d}(n)), \quad n = 1, 2, \dots, \quad (7)$$

where $\hat{d}(n)$ and $\hat{s}(n)$ are the received inter-packet delays and demodulated code symbols, respectively, at time n . In the following, we show how to obtain $T(\cdot)$ and $T^{-1}(\cdot)$.

After the encoding process, the amplitude of the code symbol $s(n)$ is a discrete random variable with a cumulative density function (CDF) denoted by $F_s(\cdot)$. Here $F_s(\cdot)$ can be calculated by accumulating the PMF of the code symbols (see Eq. (5)) as

$$F_s(l) = Pr[s \leq l] = \sum_{l_m \leq l} Pr[s = l_m] = \sum_{l_m \leq l} P_s(l_m). \quad (8)$$

Here, l_m are the unique values of code symbols sorted in ascending order.

³ Upsampling is the process of increasing the sampling rate of a signal. Normally it involves inserting $L - 1$ zeros between each sample, where L denote the upsampling factor. In our case, the upsampling factor is equal to 2.

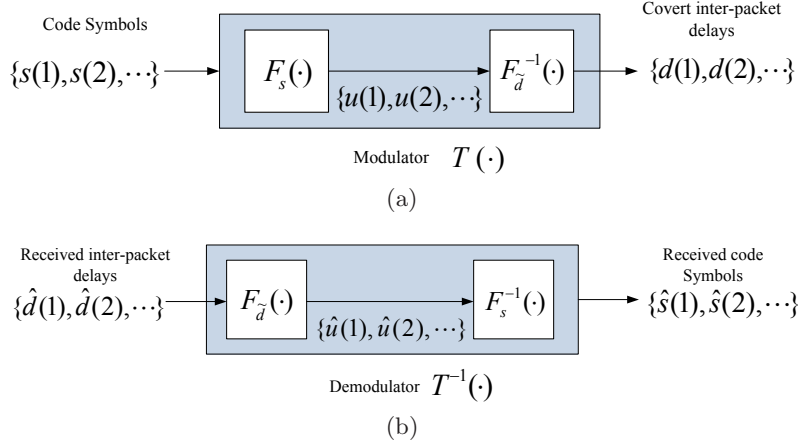


Fig. 2. Schematic description of (a) modulation and (b) demodulation processes.

To generate covert inter-packet delays following the distribution of the given i.i.d. target traffic, we employ the commonly used inverse transform technique [7]: a sequence of uniform random numbers in the range $[0, 1]$ is input to the inverse function of the CDF of the legitimate traffic. The overall two-step modulation and demodulation processes are illustrated in Figure 2. First, the random code symbols $\{s(1), s(2), \dots\}$ encoded with the covert message according to Eq. (3) are mapped to different random numbers $\{u(1), u(2), \dots\}$, which are uniformly distributed in the range $[0, 1]$. Secondly, the inverse distribution function of the legitimate traffic $F_d^{-1}(\cdot)$ takes these uniform numbers as input and generates random inter-packet delays with CDF $F_d(\cdot)$ as output. The demodulation process is very straightforward and just performs the inverse operations.

In detail the modulation process is shown in Figure 3. First, a code symbol $s(n) = l_m$ ($1 \leq m \leq M$) is input into the function $F_s(\cdot)$ to obtain its cumulative density $F_s(l_m)$. Since the CDF is monotonic, this is a 1-to-1 mapping between $s(n)$ and $F_s(l_m)$. Note that $F_s(l_m)$ is a discrete random variable in the range $(0, 1]$. Second, with the help of a CSPRNG, we generate a uniform random number $v(n)$ in the range $(0, 1]$ to achieve the randomness of the inputs for the next step. Particularly, we construct $u(n)$ by linear interpolation:

$$u(n) = F_s(l_{m-1}) + [F_s(l_m) - F_s(l_{m-1})] \cdot v(n), \quad (9)$$

where $F_s(l_0) = 0$. In fact, this corresponds to randomly picking a value $u(n)$ in the range $(F_s(l_{m-1}), F_s(l_m)]$ instead of using $F_s(l_m)$ to represent $s(n)$. It is easy to see that the overall sequence $\{u(1), u(2), \dots\}$ consists of pseudo random numbers with a uniform distribution over $(0, 1]$. Now, by letting $d(n) = F_d^{-1}(u(n))$, we obtain a random variable $d(n)$ with the CDF $F_d(\cdot)$. The resulting values $\{d(1), d(2), \dots\}$ are used as modulated inter-packet delays for the covert message $\{b_1, b_2, \dots\}$.

At the receiver, the received inter-packet delay $\hat{d}(n)$ passes through the CDF of the legitimate traffic $F_{\hat{d}}(\cdot)$ and generates $\hat{u}(n) = F_{\hat{d}}^{-1}(\hat{d}(n))$. To decode, we set $\hat{s}(n) = l_m$ if $\hat{u}(n) \in (F_s(l_{m-1}), F_s(l_m)]$. Finally, the information bits can be recovered by applying the decoding process of Eq. (4). Note that the code word \mathbf{c}_k is required in the decoding process, which is achieved by a shared codebook between the sender and receiver.

5.2 Proof of Undetectability

Theorem 1. *Consider the proposed cover timing channel mechanism and let G denote the deployed PRNG. If G is cryptographically secure with respect to a security parameter σ , then the generated covert timing channel is undetectable with respect to σ as well.*

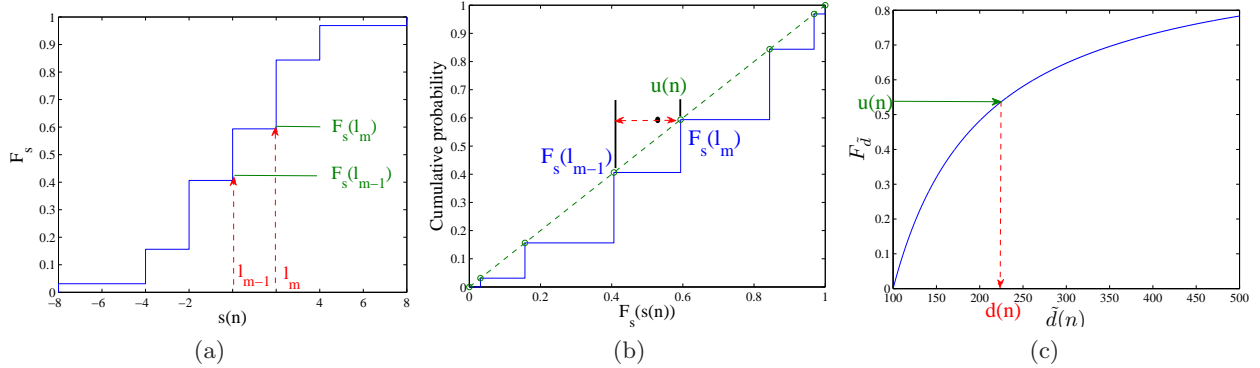


Fig. 3. An example of the mapping process (a) from $s(n) = l_m$ to $(F_s(l_{m-1}), F_s(l_m)]$, (b) from $(F_s(l_{m-1}), F_s(l_m)]$ to $u(n)$ and (c) from $u(n)$ to $d(n)$.

Proof. According to Definition 2, we have to show that $Adv_{\mathcal{D}}(\mathbb{D}_N, \tilde{\mathbb{D}}_N)$ is negligible for any distinguisher \mathcal{D} with respect to an appropriate security parameter σ where \mathbb{D}_N and $\tilde{\mathbb{D}}_N$ are the distribution of the inter-packet delays that stem from legitimate and covert traffic, respectively. For simplicity, we omit the parameter N in the following. By (9), if the values $v(n)$ are uniformly random in the range $[0, 1]$, then the values $u(n)$ are uniformly distributed in the range $[F_s(l_{m-1}), F_s(l_m)]$. Let \mathbb{U} denote the uniform distribution on R^N (with R being the range of G) and \mathbb{G} the output distribution of G . We will prove that

$$\max_{\mathcal{D}} Adv_{\mathcal{D}}(\mathbb{D}, \tilde{\mathbb{D}}) \leq \max_{\mathcal{D}'} Adv_{\mathcal{D}'}(\mathbb{U}, \mathbb{G}). \quad (10)$$

If G is cryptographically secure, then the right-hand side of (10) is negligible by definition. In consequence, $\max_{\mathcal{D}} Adv_{\mathcal{D}}(\mathbb{D}, \tilde{\mathbb{D}})$ is negligible as well, being exactly the condition for undetectability.

Let \mathcal{D} denote an arbitrary distinguisher between \mathbb{D} and $\tilde{\mathbb{D}}$.

We turn \mathcal{D} into a distinguisher between \mathbb{U} and \mathbb{G} that has the same effort and advantage as \mathcal{D} . This will prove Eq. (10). The distinguisher \mathcal{D}' is defined as follows. Input are some values $\mathbf{u} = (u(1), u(2), \dots, u(N))$ with either $\mathbf{u} \leftarrow \mathbb{D}$ or $\mathbf{u} \leftarrow \mathbb{G}$. The distinguisher \mathcal{D}' uses this input to create some covert traffic as described in Section 5; this results in some inter-packet delays $\mathbf{d} = (d(1), d(2), \dots, d(N))$ that are handed to \mathcal{D} . Distinguisher \mathcal{D}' receives a bit output $b \in \{0, 1\}$ from \mathcal{D} and uses it as its own output.

Now observe that if $\mathbf{u} \leftarrow \mathbb{G}$, i.e., is a PRNG-sequence, then \mathcal{D}' has correctly generated covert traffic and it holds that $\mathbf{d} \leftarrow \tilde{\mathbb{D}}$. On the other hand, if $\mathbf{u} \leftarrow \mathbb{U}$, i.e., the sequence is truly random, then the generated traffic is truly i.i.d. as in the legitimate traffic, i.e., $\mathbf{d} \leftarrow \mathbb{D}$. This shows that

$$Adv_{\mathcal{D}}(\mathbb{D}, \tilde{\mathbb{D}}) = Adv_{\mathcal{D}'}(\mathbb{U}, \mathbb{G}), \quad (11)$$

which in turn implies (10) and, as explained above, in particular the claim. \square

5.3 Determining Encoding Parameters

According to the Definition 3, the system robustness is measured by the ratio of SNR before and after performing the encoding and modulation process. To generate undetectable covert traffic, the core idea is to generate random numbers uniformly distributed in the range $[0, 1]$. There are two methods to achieve this goal. After the encoding process, a code symbol $s(n) = l_m$ will be

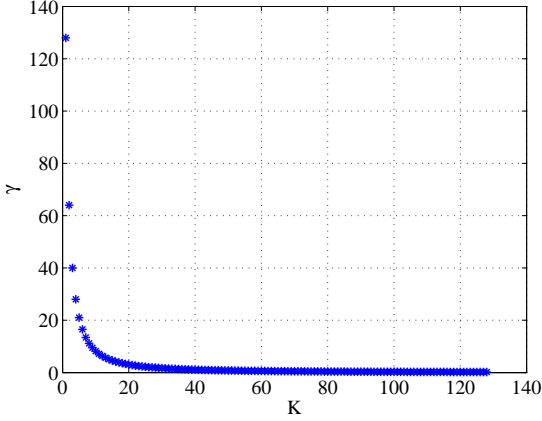


Fig. 4. The impact of K on the parameter γ ($N = 128$).

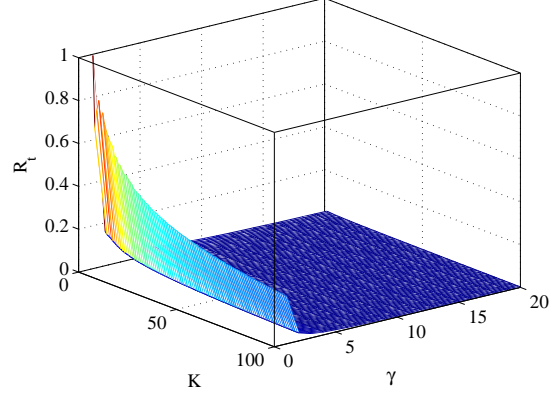


Fig. 5. Transmission rate R_t vs. the number of orthogonal channels K , and robustness gain γ .

converted into a pseudo uniform random number $u(n)$ within the interval $(F_s(l_{m-1}), F_s(l_m)]$, where there are M different possible ranges in total. On the other hand, without the above encoding process, $s(n)$ is just the original information bit with equal probabilities. To generate uniformly distributed numbers, we map $s(n)$ into the a number $u_1(n)$, which is uniformity distributed in the range $(0, 1/2]$ and $(1/2, 1]$, for the bit-0 and bit-1, respectively. The robustness gain is equivalent to the signal power ratio between $u(n)$ or $u_1(n)$ for the same information bits, provided the noise are the same for both the cases. Given Definition 3 and the spreading factor N from the encoding process, the robustness gain can be calculated by $\gamma = N \cdot E_K/E_1$, where E_K is the power of the random variable $u(n)$ when K channels are used and E_1 is the power of the $u_1(n)$. Note that, for a uniform random variable in $(a, b]$, its power is represented by the variance, that is, $(b - a)^2/12$. Considering that there are M different possible ranges for $u(n)$ with equal probability, the total robustness gain γ is given by

$$\gamma = \frac{N}{M} \sum_{m=1}^M \left[\frac{1}{2} \frac{E_K(u(n) = l_m)}{E_1(u_1(n) = 1)} + \frac{1}{2} \frac{E_K(u(n) = l_m)}{E_1(u_1(n) = -1)} \right] = \frac{N}{M} \sum_{m=1}^M \frac{[F_s(l_m) - F_s(l_{m-1})]^2}{(\frac{1}{2})^2} = \frac{4N}{M} \sum_{m=1}^M [P_s(l_m)]^2. \quad (12)$$

Given the distribution of $s(n)$ in Eq.(5) and $M = K + 1$, we rewrite Eq. (12) as:

$$\gamma = \frac{4N}{M} \sum_{m=1, -K \leq l_m \leq K}^M \left(\frac{K}{\frac{K-l_m}{2}} \right)^2 \left(\frac{1}{2} \right)^{2K} = N \cdot \frac{1}{K+1} \left(\frac{1}{2} \right)^{2K-2} \sum_{k=0}^K \binom{K}{k}^2. \quad (13)$$

The effect of K on the gain γ is illustrated in Figure 4. With N fixed, it is easy to verify that γ is monotonically decreasing with K . At the same time, since γ is a linearly increasing function of N , one can achieve a higher robustness by decreasing K and increasing N . Therefore, any given gain requirement γ_0 can be achieved by choosing appropriate values for K and N . A reasonable choice of K and N is to maximize the data transmission rate $R_t = K/N$. Next we present a solution to achieve this goal. Eq. (13) can be expressed as $\gamma = N \cdot q(K)$ by setting

$$q(K) = \frac{1}{K+1} \left(\frac{1}{2} \right)^{2K-2} \sum_{k=0}^K \binom{K}{k}^2. \quad (14)$$

To achieve a desired robustness gain requirement γ_0 , the spreading gain N needs to satisfy $N \geq \gamma_0/q(K)$. Since N is an integer, we choose $N = \lceil \gamma_0/q(K) \rceil$, which is the smallest value that satisfies the robustness requirement. Finally, we choose the best K to maximize $R_t = K/\lceil \gamma_0/q(K) \rceil$.

Figure 5 shows R_t as a function of K and γ_0 . It shows that the best K for all values of γ occurs when $K = 1$. This also corresponds to the maximum robust gain γ , as mentioned above. From Eq. (13), $\gamma = N$ and the transmission rate R_t becomes $1/N$. Therefore, by setting $K = 1$ and $N = \lceil \gamma_0 \rceil$, our system can achieve any desired robustness requirement γ_0 , corresponding to a specific BER P_e .

5.4 Algorithm Summary

The function *InterPacketDelayGenerator*($\gamma_0, F_{\bar{d}}$) in Algorithm 6.1 describes how to generate the covert inter-packet delays \mathbf{d} under given robustness and security requirements. Since only one channel is used to transmit the covert message, a code symbol $s(n)$ is also binary and its CDF must satisfy $F_s(-1) = \frac{1}{2}$ and $F_s(1) = 1$. Equivalently, we have $F_s(s(n)) = \frac{1}{2} + \frac{1}{4}(1 + s(n))$. Note that if only one fixed channel is utilized during the encoding process, it is fairly easy for an adversary to obtain the code word. To prevent him from decoding the received code symbols, in our algorithm, we dynamically change the index k of the channel (i.e., the spreading code word).

6 Experimental Results

In this section, we validate the effectiveness of our covert timing channel through a series of experiments.

6.1 Experimental Setup

Based on our design, we have developed a covert timing channel testbed running over TCP/IP networks. The testbed consists of a server and a client which act as the sender and the receiver of both the covert and the overt communication. We insert the inter-packet delays for covert channel modulation or additional noise using hooks in the Linux network stack at the sender. The receiver passively collects the packet inter-arrival delays using tcpdump and decodes them with the shared code book and a shared key.

Test Scenarios. Two test scenarios are considered in our experimental evaluation. The first scenario is in a LAN environment in the campus network of a US university; client and server are hosted at two different departments. The connection passes through several switches, routers and firewall devices. The second scenario is in the WAN environment to represent worldwide Round Trip Times (RTTs) between sender and receiver. The sender is hosted in the US and the receiver in Germany. The network attributes of the two experimental scenarios are summarized in Table 1. Here the packet retransmission rate is measured using the ping command. We compute the jitter statistics based on the difference of delays between packets leaving the source and arriving at the destination.

Dataset. To test our covert channel for i.i.d. traffic, we adopt Telnet traffic using TCP/IP transmission as the medium for the covert timing channel. Paxson and Floyd [18] have shown that the inter-packet delays of a Telnet session (from the client to the server) can be well modeled by an i.i.d. Pareto distribution. Specifically, its CDF can be defined as

$$F_{\bar{d}}(s) = 1 - \left(\frac{\alpha}{s}\right)^\beta, \quad (15)$$

Algorithm 6.1: INTERPACKETDELAYGENERATOR($\gamma_0, F_{\bar{d}}$)

Input : robust gain γ_0 , distribution of legitimate inter-packet delays $F_{\bar{d}}(\cdot)$

Output : covert inter-packet delays \mathbf{d}

$N \leftarrow \lceil \gamma_0 \rceil$ // estimate spreading ratio with given robustness gain

for each information bit b

do $\left\{ \begin{array}{l} \text{generate } k \in_{\mathbb{N}} \text{Uniform}[1, N] \\ (s(1), \dots, s(N)) \leftarrow b \cdot \mathbf{c}_k \quad // \text{encoding} \\ \text{generate } (v(1), \dots, v(N)) \in_{\mathbb{R}} \text{Uniform}(0, 1] \\ u(n) \leftarrow F_s(s(n)) - \frac{1}{2} + \frac{1}{2}v(n) \quad // \text{modulation } s \text{ to } u \\ d(n) \leftarrow F_{\bar{d}}^{-1}(u(n)) \quad // \text{modulation } u \text{ to } d \\ \mathbf{d} := (d(1), d(2), \dots, d(N)) \end{array} \right.$

Table 1. The network conditions for each test scenario.

	LAN	WAN
Physical distance (miles)	1.5	5352
Packet retransmission rate (%)	0.60	1.03
Jitter(std) (ms)	12.6814	25.4023
Jitter(mean)(ms)	0.0550	0.10274

where $s > \alpha$. Here, $\alpha \in \mathbb{R}^+$ is a scale parameter and $\beta \in \mathbb{R}^+$ is a shift parameter. For illustrative purposes, the legitimate samples that we use for our experiments were generated using the above mentioned theoretical distribution function. Similar to [22], our model parameters are set as $\alpha = 100$ ms and $\beta = 0.95$, which are the traffic model parameters obtained from real Telnet traffic [18]. The total number of legitimate traffic is 1,000,000 inter-packet delays.

Note, like most existing encoding schemes [21], the sequence number information obtained from analyzing the TCP packet header is used to align the encoded traffic for correct decoding. This will ensure that there is no error propagation. Specifically, errors occurring earlier will not affect the decoding capability of a covert message sent later.

6.2 Performance Analysis

In our experiments, we generate a sequence of covert packets to carry 100,000 random information bits. The channel undetectability, robustness and the transmission tradeoff is discussed in the following.

Undetectability. Although we have already provided a proof that our i.i.d. traffic based covert timing channel is undetectable against any polynomial distinguisher (see Section 5.2), we use some standard statistical tests to visually show and verify undetectability. We first use the Quantile-Quantile (Q-Q) plot to visually examine the statistical similarity of inter-packet delays between the covert traffic generated by our proposed method and the legitimate traffic of a Telnet connection (see Figure 6). In a Q-Q plot, if the two sets have the same distribution, the points should fall approximately along a reference line. We observe that the distribution of our covert traffic and legitimate traffic have an excellent match. This conclusion can be further verified using the Kolmogorov-Smirnov test (KS-test) [2], which measures the maximum distance H_s of samples between the legitimate traffic and covert traffic. Our results show that the maximum H_s between the

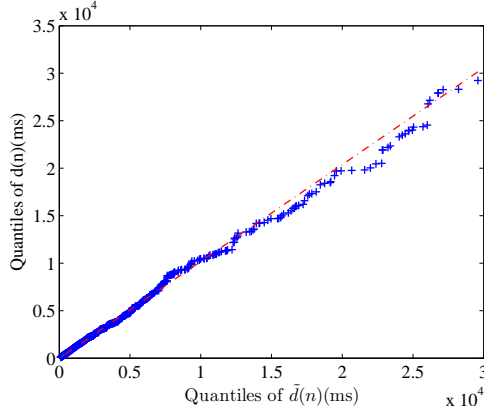


Fig. 6. $Q - Q$ plot of the inter-packet delay of covert traffic $d(n)$ vs the inter-packet delay of legitimate traffic $\tilde{d}(n)$ for Telnet application.

covert traffic and the legitimate traffic is 0.0144 while the minimum value among all pairs of legitimate traffic samples is 0.028. Therefore, the covert traffic can be considered undetectable under the KS-test.

Robustness. To evaluate the robustness of the proposed algorithm, we consider three different types of noise during the transmission process. The first type represents the inherent network noise due to packet loss, delay, and jitter. The second and the third types of noise are jamming noises with different distributions which may be injected by an active adversary. Specifically, we use noise with a normal distribution with zero mean and variance σ^2 as second noise type. A uniformly distributed noise represents the worst case scenario in terms of channel capacity⁴ [21], as a random variable with uniform distribution has maximum entropy among all random variables over a fixed range. Therefore, we choose the third type of noise uniformly distributed in the range $[0, \Delta]$. Note that, as long as the spreading codes used in the encoding process are orthogonal to $(1, 1, \dots, 1)$, the mean of the noise does not impact the demodulation.

Table 2 summarizes the robustness test results for the overall system. The average inter-packet delays of legitimate traffic is around 212 ms. In each experiment, covert inter-packet delays are generated with a given robustness requirement. This requirement is defined by the robustness gain γ_0 . In our test, we set γ_0 as 1, 5 and 10 to illustrate the effect of an absent, a moderate and a strong spreading factor. In addition, we compare the robustness performance between our scheme and the (L, n) undetectable covert timing channel of [22]. Particularly, we choose the $(8, 2)$ scheme due to its high transmission rate. We also select the $(1, 1)$ scheme which is the most robust (L, n) scheme as a result of the largest range of each code symbol (the total range for all code symbols is fixed in $[0, 1]$), which ensures the largest signal power, so as the system SNR.

From these results we observe that when there is no jamming noise, there are no bit errors in the LAN scenario. When there is no spread encoding ($N = 1$), Gaussian noise with $\sigma = 50$ ms can lead to more than 4.04% errors and the noise uniformly distributed between $[0, 50]$ ms may lead to up to 3.61% errors. It is very similar to the performance of the $(1, 1)$ encoding scheme, which determines the upper bound of the robustness of the (L, n) encoding scheme. However, once increasing the robustness gain to 5, the correct bit rate $(1 - P_e)$ achieved by our proposed algorithm is more than 97.34% for both the LAN and WAN tests for jamming noise range of $[0, 100]$ ms. When γ_0 increases to 10, the correct bit rate can achieve more than 99.95% for additive Gaussian noise with $\sigma = 200$

⁴ Channel capacity is the maximum information that can be reliably transmitted over a communications channel.

Table 2. Summary of the bit error rate P_e (%) for the timing channel experiments.

Test scenario	Encoding scheme	Channel noise	Gaussian σ (ms)				Uniform Δ (ms)				
			50	100	150	200	20	50	100	150	
LAN	γ_0 spreading	1	0	4.04	6.06	11.23	13.60	0	3.21	8.42	12.63
		5	0	0	0.24	0.63	1.08	0	0	1.80	4.84
		10	0	0	0	0	0.02	0	0	0.004	0.56
	(L, n) encoding	(1, 1)	0	4.50	8.13	10.63	13.00	0	3.00	8.25	12.96
		(8, 2)	0	10.13	14.06	14.69	15.76	5.12	11.53	14.19	15.69
WAN	γ_0 spreading	1	0.02	8.22	15.43	18.04	21.24	2.71	3.61	10.62	16.83
		5	0.005	0.01	0.55	1.20	4.04	0.01	0.23	2.66	5.96
		10	0	0	0.002	0.01	0.05	0	0.002	0.08	0.72
	(L, n) encoding	(1, 1)	0.01	4.75	9.38	11.50	13.13	0.15	3.63	8.50	13.50
		(8, 2)	0.06	11.27	14.16	15.72	16.66	5.53	11.94	14.25	15.75

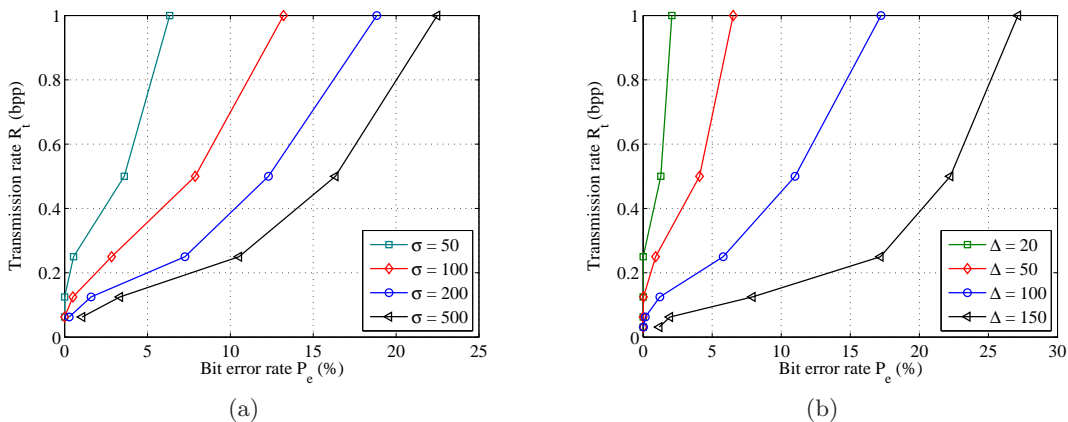


Fig. 7. Trade-off between the transmission rate R_t and the bit error rate P_e under (a) Gaussian and (b) uniform noise.

ms. Even when the upper limit of uniform noise is increased to 150 ms, we can still correctly transmit more than 99.28% of the total bits. These results show dramatic BER improvements compared to the (8, 2) encoding scheme, which can achieve a good data rate.

Trade-off. By analyzing the results obtained in the LAN and WAN scenarios and comparing them to the (L, n) encoding scheme [22], we have shown that increasing the spreading ratio N can significantly reduce the system BER P_e . However, this also requires more inter-packet delays in the encoding process to deliver one information bit and thus reduces the transmission rate R_t . To investigate their relationship, Figure 7 plots the transmission rate R_t versus P_e under different noise scenarios in the LAN environment. It clearly shows that there is a trade-off between the transmission rate R_t and the robustness. In particular, the bit error rate increases monotonically with the transmission rate. This property can easily be verified by examining the definition of R_t , which is $1/N$, and the measure of robustness gain N . Therefore, our system provides a solution for balancing the system robustness and transmission rate. At the same time, we note that the achieved security level is independent of transmission rates and robustness.

7 Conclusion and Future Work

We propose a method to modulate a covert timing channel on network traffic that constitute independent and identically distributed (i.i.d.) inter-packet delays. Our covert timing channel is

both robust and provably undetectable (i.e., no polynomial adversary can distinguish the legitimate traffic from the covert one). We discuss the choice for i.i.d. traffic and confirm the theoretical results through variety of measurements on synthetic Telnet traffic from network modeling. Since i.i.d traffic models are building blocks of more complex traffic models, a natural open question is the extension of our approach for real applications such as video streaming or Voice over IP (VOIP).

References

1. V. Berk, A. Giant, and G. Cybenko. Detection of covert channel encoding in network packet delays. *Technique Report, Dartmouth College*, 2005.
2. D. C. Boes, F. A. Graybill, and A. M. Mood. *Introduction to the Theory of Statistics, 3rd ed.* New York: McGraw-Hill, 1974.
3. S. Cabuk. Network covert channels: Design, analysis, detection, and elimination. *PhD thesis, Purdue University*, 2006.
4. S. Cabuk, C. E. Brodley, and C. Shields. IP covert timing channels: design and detection. In *CCS '04: Proceedings of the 11th ACM Conference on Computer and Communications Security*, pages 178–187, New York, 2004.
5. S. Cabuk, C. E. Brodley, and C. Shields. IP covert channel detection. *ACM Transaction of Information System and Security*, 12(4):1–29, 2009.
6. J. Cao, W. S. Cleveland, D. Lin, and D. X. Sun. Internet traffic tends to poisson and independent as the load increases. *Technical Report, Bell Labs*, 2001.
7. L. Devroye. *Non-Uniform Random Variate Generation*. New York: Springer-Verlag, 1986.
8. V. Frost and B. Melamed. Traffic modelling for telecommunications network. *IEEE Communication Magazine*, 32(3):70–80, March 1994.
9. S. Gianvecchio and H. Wang. Detecting covert timing channels: An entropy-based approach. In *CCS '07: Proceedings of the 14th ACM Conference on Computer and Communications Security*, pages 307–316, 2007.
10. S. Gianvecchio, H. Wang, D. Wijesekera, and S. Jajodia. Model-based covert timing channels: Automated modeling and evasion. In *RAID '08: Proceedings of the 11th International Symposium on Recent Advances in Intrusion Detection*, pages 211–230, Berlin, Heidelberg, 2008. Springer-Verlag.
11. J. Giles and B. Hajek. An information-theoretic and game-theoretic study of timing channels. *IEEE Transactions on Information Theory*, 48(9):2455–2477, 2002.
12. H. Hefes and D. Lucantoni. A markov modulated characterization of packetized voice and data traffic and related statistical multiplexer performance. *IEEE Journal on Selected Areas in Communications*, 4:856–868, Sep. 1986.
13. L. Kleinrock. *Queueing Systems*. Wiley New York, 1976.
14. W. Li, R. J. Fretwell, and D. D. Kouvatsos. Analysis of correlated traffic by batch renewal process. In *EBISS '09: International Conference on E-Business and Information System Security*, pages 1–5, June 2009.
15. Y. Liu, D. Ghosal, F. Armknecht, A. Sadeghi, S. Schulz, and S. Katzenbeisser. Hide and seek in time - robust covert timing channels. In *ESORICS '09: the 14th European Symposium on Research in Computer Security*, volume 5789 of *Lecture Notes in Computer Science*, pages 120–135. Springer, 2009.
16. D. M. Lucantoni, M. F. Neuts, and A. R. Reibman. Methods for performance evaluation of VBR video traffic models. *IEEE/ACM Transactions on Networking*, pages 176–180, Apr. 1994.
17. M.A. Padlipsky, D.W. Snow, and P.A. Karger. Limitations of end-to-end encryption in secure computer networks. *Technical Report ESD TR-78-158, Mitre Corporation*, 1978.
18. V. Paxson and S. Floyd. Wide area traffic: the failure of poisson modeling. *IEEE/ACM Transactions on Networking*, 3(3):226–244, 1995.
19. R. Prasad and S. Hara. An overview of multi-carrier CDMA. In *IEEE the 4th International Symposium on Spread Spectrum Techniques and Applications Proceedings*, volume 1, pages 107–114, 1996.
20. J. Proakis. *Digital Communications*. Singapore: McGraw Hill, 1995.
21. S. H. Sellke, C. Wang, N. Shroff, and S. Bagchi. Capacity bounds on timing channels with bounded service times. In *IEEE International Symposium on Information Theory*, pages 981–985, 2007.
22. S.H. Sellke, C. Wang, S. Bagchi, and N. Shroff. TCP/IP timing channels: Theory to implementation. In *INFO-COM '09: IEEE Conference on Computer Communications*, pages 2204–2212, April 2009.
23. G. Shah, A. Molina, and M. Blaze. Keyboards and covert channels. In *USENIX-SS'06: Proceedings of the 15th Conference on USENIX Security Symposium*, pages 59–75, 2006.
24. S. Zander, G. Armitage, and P. Branch. A survey of covert channels and countermeasures in computer network protocols. *IEEE Communications Surveys & Tutorials*, 9(3):44–57, 2007.