Hindawi
Security and Communication Networks
Volume 2020, Article ID 8892896, 20 pages
https://doi.org/10.1155/2020/8892896

*Review Article*

# A Survey of Key Technologies for Constructing Network Covert Channel

**Jing Tian**,[1,2] **Gang Xiong**,[1,2] **Zhen Li**,[1,2] **and Gaopeng Gou**[1,2]

[1]*Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China*
[2]*School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China*

Correspondence should be addressed to Gaopeng Gou; gougaopeng@iie.ac.cn

Received 8 April 2020; Accepted 16 July 2020; Published 5 August 2020

Academic Editor: Leonardo Mostarda



In order to protect user privacy or guarantee free access to the Internet, the network covert channel has become a hot research topic. It refers to an information channel in which the messages are covertly transmitted under the network environment. In recent years, many new construction schemes of network covert channels are proposed. But at the same time, network covert channel has also received the attention of censors, leading to many attacks. The network covert channel refers to an information channel in which the messages are covertly transmitted under the network environment. Many users exploit the network covert channel to protect privacy or guarantee free access to the Internet. Previous construction schemes of the network covert channel are based on information steganography, which can be divided into CTCs and CSCs. In recent years, there are some covert channels constructed by changing the transmission network architecture. On the other side, some research work promises that the characteristics of emerging network may better fit the construction of the network covert channel. In addition, the covert channel can also be constructed by changing the transmission network architecture. The proxy and anonymity communication technology implement this construction scheme. In this paper, we divide the key technologies for constructing network covert channels into two aspects: communication content level (based on information steganography) and transmission network level (based on proxy and anonymity communication technology). We give an comprehensively summary about covert channels at each level. We also introduce work for the three new types of network covert channels (covert channels based on streaming media, covert channels based on blockchain, and covert channels based on IPv6). In addition, we present the attacks against the network covert channel, including elimination, limitation, and detection. Finally, the challenge and future research trend in this field are discussed.

## 1. Introduction

With the rapid development of information technology, Internet has penetrated into every aspect of people's lives. However, when people enjoy the convenience brought by the network, there have been many issues of information leakage and user privacy breaches [1]. For example, there have emerged malicious attacks which aimed at stealing confidential government data, such as GhostNet [2], ShadowNet [3], and Axiom [4]. On the other hand, repressive governments have deployed increasingly sophisticated technology to block the disfavored Internet content [5]. So, many users cannot access Internet freely.

The network covert channel can covertly transmit secret messages. It can hide covert traffic in a large amount of overt communication traffic. Many researches show that the use of network covert channel can protect user privacy and guarantee users' right to free access to Internet [6–8]. The secure transmission of secret messages in the communication process refers to two aspects: one is the communication content security [9] and the other is the communication connection security [6, 10, 11]. Network covert channel can effectively improve the security of these two aspects.

In terms of communication content security, encryption technology is widely used to protect the communication content of both sides, such as SSL (secure sockets layer),

digital signature, and other technologies. The Google transparency report "HTTPS Encryption in Chrome" (available under https://transparencyreport.google.com/https/overview) states that, in October 2019, 95% of Chrome webpages enabled encryption. In addition, according to Netmarketshare (a website for Market Share Statistics for Internet Technologies, available under https://netmarketshare.com/report.aspx?id=https), the percentage of encrypted web traffic in October 2019 has exceeded 90%.

However, with the continuous development of the encrypted traffic analysis technology, even in the case of encryption, certain activities of users can still be discovered [12, 13]. So, the privacy of users cannot be well protected. On the other hand, the increasing computing power and attacks on encryption algorithm also make it possible to crack encrypted traffic [14, 15]. The covert channel can prevent the encrypted traffic from being discovered due to its covert transmission characteristics. In this environment, if the attacker does not know the covert channel construction method, he cannot perform the attacks on encrypted traffic, even if he has a strong ability to analyze and crack encrypted traffic [7]. So, the network covert channel enhanced the communication content security.

In terms of communication connection security, the meta-data (message source IP address, destination IP address, etc.) and communication mode (interval of packets, etc.) cannot be hidden by encryption [10]. The communication participants may expose identity information to the network eavesdroppers [16]. Further, they can infer the sender and receiver of the message and find the ongoing communication connection, leading to significant risk of privacy leaks and being blocked.

But, the network covert channel is an unconventional communication method, and the eavesdroppers cannot determine whether the user is actually performing covert communication and thus cannot find both sides of communication. So, the identity concealment of both parties can be protected [17]. On the other hand, because the traffic of the covert channel is mixed in a large amount of overt traffic, even if the eavesdroppers use some methods to obtain the identity of both parties, it is difficult for them to determine whether the two parties are sending or receiving messages, that is, the communication behavior is unobservable [18]. So, the covert channel can provide a strong guarantee for the security of communication connection.

The use of covert channels strengthens the content security of encrypted traffic and fills the shortcomings that encryption cannot protect the security of communication connection. So, the demand to construct network covert channels is increasing, and many technologies are proposed. The most common technology is to use information steganography to build a network covert channel [17]. The information steganography can hide secret messages in the temporal behavior of the traffic or the storage fields in the network protocol, which composes CTCs (covert timing channels) and CSCs (covert storage channels) accordingly [7]. Besides the information steganography, many covert channels perform covert transmission by changing the transmission network architecture. There are two typical representatives: proxy technology [19, 20] and anonymous communication technology [11]. The proxy can be divided into two categories: end-to-end proxy (such as HTTP proxy [21]) and end to middle proxy (such as Telex [22]). In addition, anonymous communication technology can also conduct a new covert transmission path. There are many mature anonymous communication systems such as Tor [23], I2P [24], and Loopix [11].

On the other side, some research work promises that the characteristics of emerging networks may better fit the construction of the network covert channel. With the development of emerging networks, many network covert channels in the new network environment (streaming media network, blockchain network, and IPv6) have been proposed. The covert channels based on streaming media network hide secret messages in audio and video traffic and use popular streaming media applications as the carrier. There are three typical covert channels: Facet [25], CovertCast [26], and DeltaShaper [27]. The blockchain network has the characteristics of participant-anonymity, flooding propagation, and tampering resistance [28]. The covert channels based on blockchain network can utilize participant-anonymity and flooding propagation to increase the concealment of communicating parties. The tampering resistance can also be used to guarantee the robustness of covert channel. In this context, the models of covert channels based on blockchain network are proposed [10, 28] and three covert channels (Zombiecoin [10], Botchain [29], and Chainchannels [30]) have been actually deployed in blockchain network. The IPv6 network is also a compelling platform for constructing covert channels. The IPv6 header and its extensions have many reserved fields or other fields which can embed information, thus leading to many possible covert channels [31].

However, because the network covert channel is a good method to cope with repressive government, it has also received the attention of censors [32]. Compared with ordinary eavesdroppers, the national-level censors have a global traffic view and have a stronger ability to analyze traffic. More and more attacks against the covert channel have appeared, which has an impact on channel concealment, robustness, and transmission efficiency [33–35].

Although there are many studies on covert channels, there is no comprehensive survey for the construction technologies they use and corresponding attacks. In addition, there is also less research on the covert channels in the new network environment. Compared with the already published studies, the main contributions of this paper are as follows:

(1) Previous studies only considered the network covert channel based on information steganography, but not the covert channel based on the changing network architecture. According to different principles of covert channel construction technologies, we divide covert channels into two levels: communication content and transmission network, which can comprehensively include existing covert channels. And, we conduct a comprehensive analysis on the covert channels under each construction technology.

(2) The characteristics of the new network create many convenient conditions for the construction of network covert channels. However, they are not considered in other reviews. We present the covert channels in the new network environments including streaming media, blockchain, and IPv6, which makes up for deficiencies in existing work. It would highly facilitate for the researchers to understand the research status and provide research ideas for the subsequent design of covert channels in those new network environments.

(3) We emphasize the challenging problems facing the construction of covert channels: the IP blocking or other blocking technology reduces the channel availability; the use of ML and DL technology makes the covert channel easier to expose. We discuss how to improve the ability to resist those problems, such as using adversarial examples, constructing reversible network, covert channel.

In order to improve the readability, we list the abbreviations used in our article in Table 1.

The rest of the paper is organized as follows: Section 2 gives the research background of network covert channels. In Section 3, we present the network covert channel construction technology at communication content level and transmission network level. In Section 4, we provide the covert channels in the new network environment. In Section 5, we present network covert channel metrics including concealment, robustness, and throughput. In Section 6, we show the attacks against network covert channels. Then, we discuss the challenges and suggest future research directions in Section 7. Lastly, Section 8 presents conclusion.

Table 1: The abbreviations used in our article.

| Abbreviation | Full name |
|---|---|
| SSL | Secure sockets layer |
| Tor | The second-generation onion router |
| DPI | Deep packet inspection |
| I2P | Invisible internet project |
| ML | Machine learning |
| DL | Deep learning |
| CTCs | Covert timing channels |
| CSCs | Covert storage channels |
| URL | Uniform resource locator |
| E2M | End-to-middle |
| C&C | Command and control |
| IPDs | Internet packet delays |
| BER | Bit error rate |
| PDU | Protocol data unit |
| ICMP | Internet control message protocol |
| ECDH | Elliptic curve Diffie-Hellman |
| TCP ISNs | TCP initial sequence numbers |
| PPTP | Point-to-point tunneling protocol |
| L2TP | Layer two-tunneling protocol |
| VTP | VLAN trunking protocol |
| IPSec | IP security |
| DHT | Distributed hash table |
| ESP | Encapsulating security payload |
| SDN | Software defined network |
| IoT | Internet of things |
| ICS | Industrial control systems |
| DGA | Domain generation algorithm |

## 2. Research Background of Network Covert Channels

*2.1. Network Covert Channel Definition.* The covert channel was originally proposed by Lampson [36]. Its purpose is to transmit secret messages to the recipient in an unconventional manner without being noticed by the observer.

The classic communication scenario of the covert channel is the prisoner problem [37]: Alice and Bob are held in two rooms of a prison. They want to escape and they need to transmit the escape plan to each other, but the watcher Wendy monitors them. Therefore, Alice and Bob need to complete the information exchange without alerting Wendy.

With the development of Internet, the network covert channel has emerged. It is a kind of channel that transmits covert messages in violation of communication restriction rules in network environment [7]. The goal of network covert channels is not only to ensure that communication content is not discovered but also to protect the identities of both parties. The prisoner model of the network covert channel is shown in Figure 1.

*2.2. Adversary Scenario.* In order to improve the availability of covert channel, we must describe the adversaries appropriately. Common attack methods used by adversaries are described in detail in Section 6. There are four attributes which can describe different types of adversaries:

(i) According to the attack mode, the adversaries are divided into passive and active: the passive adversary observes the communication traffic and analyzes it; the active adversary not only observes and analyzes it but also can generate, modify, delete, or delay traffic.

(ii) According to the location, the adversaries are divided into external and internal: external adversaries are located outside the path of covert information transmission and internal attackers are located on the path or control some middleware on the path.

(iii) According to the adversaries' resource, the adversaries are divided into global and partial: a global adversary can observe the total channel, but a partial adversary can only attack a part of the channel.

(iv) According to the variability of attackers' resource, the adversaries are divided into invariable and adaptive: an invariable adversary cannot change resources he occupied after the attack begins. But, adaptive adversaries may constantly change the occupied resources during the attack.

Because of the differences in security targets and features of each covert channel, the adversary each channel is assuming is different. An adversary is always assumed to have

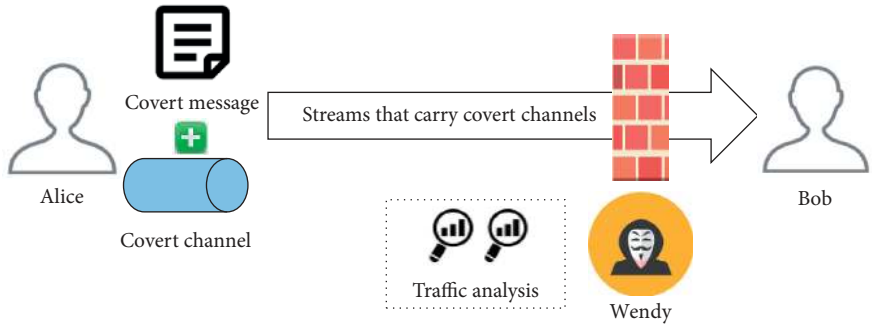


Figure 1: The prisoner model for network covert channel. Alice and Bob are the two communication parties. They encode the covert message into the covert channel, and Wendy is the observer, monitoring and analyzing the communication traffic between them.

the passive capability when analyzing the security of covert channel. But, at the same time, he may have many other attributes. For example, in Telex [22], an end-to-middle proxy, the adversary is assumed to be a person who controls the infrastructure of the network within its jurisdiction and can observe, alter, block, or inject network traffic. So, he is not only a passive adversary but also a partial and active adversary. In Tor [23], a low-latency anonymous communication system, if an adversary operates some onion nodes and modifies the data that flow through them, he is an internal and active adversary. In the meantime, if he observes all of the onion nodes, he is also a global and passive adversary.

*2.3. Network Covert Channel Classification.* Most of the existing researches divide the network covert channels into CTCs and CSCs according to the different message modulation methods [38, 39]:

(i) Network covert storage channels: CSCs include the secret messages into storage objects at the sender and then reading them at the receiver. Reserved bits or unused bits of the protocol are mainly used to transmit information.

(ii) Network covert timing channels: CTCs include the secret messages into the timing behavior at the sender and then extract the covert messages at the receiver. Normally, the delays in network packets are used to deliver covert messages.

## 3. Typical Techniques for Constructing Network Covert Channels

Most covert channels tend to encode covert information into a storage field or time behavior and then transmit it in accordance with common network transmission processes. It is essentially a form of information hiding technology (also called information steganography). The classification method mentioned in Section 2 is also based on this technology. The information steganography is a hidden transmission technology at the level of communication content, which does not involve the transmission network level. At present, there are many covert channels based on changing the structure of the transmission network. They covertly transmit information by designing new network transmission paths. So, the classification mentioned in Section 2 can no longer include all network covert channel construction schemes. We glean the covert channel researches and make a close reading of the study of techniques they use. In order to make a comprehensive summary, we divide the construction technology as follows:

(i) Covert channel construction technology at the communication content level

(ii) Covert channel construction technology at the transmission network level

Correspondingly, covert channels can also be divided into communication content level and transmission network level. In Section 3.1, we will introduce the two types of technologies and the typical covert channels under each construction technology in detail.

*3.1. Construction Technology at the Communication Content Level.* As we stated above, the construction technology at the communication content level is based on information steganography, which includes CTC construction technology and CSC construction technology.

The covert channels using this construction technology transmit secret messages through information steganography. It can be divided into CTC construction technology and CSC construction technology.

*3.1.1. CTCs.* A large body of literatures deals with the study [17, 39–41]. Wendzel et al. [39] use pattern language markup language (PLML) to classify covert channels into 11 different patterns. We refer to the classification method proposed by Wendzel et al. [39] and divide CTCs construction technology into four categories: interarrival time (C1), rate modulation (C2), PDU order modulation (C3), and PDU retransmission (C4).

*(1) Interarrival Time (C1).* The CTC construction technology based on interarrival time (C1) is the most common. It transmits messages by altering timing intervals between network PDUs. Most studies on CTCs are based on this.

Cabuk et al. proposed On-Off CTC [42], which is a time window-based method. The sender and receiver share a time

window $T_w$. When transmitting information, the total transmission time is divided into equal and disjoint time intervals $T_i$ according to the $T_w$. In a $T_i$, if the sender transmitted a packet, it represents the bit "1;" if the sender remained silent during $T_i$, it represents the bit "0."

Shah et al. [43] proposed the Keyboard Jitterbug. In Jitterbug CTC, the sender and receiver share a value $w$. The sender sends a packet with extra delay to the server when the user types the keyboard. If the delay is an integer multiple of $w$, it represents the bit "1;" if the delay is an integer multiple of $(w/2)$, it represents the bit "0." In addition, the Jitterbug CTCs do not require the sender and receiver to keep time in sync, just that their respective clocks are accurate.

With the development of steganography technology, the more concealed statistical-based CTCs have appeared. Brodley and Spafford [44] proposed the TRCTC (time-replay CTC). It sorts the overt network packet intervals and records them in the set $S_0$ and $S_1$, respectively. When transmitting bit "1," the sender randomly selects a packet interval from set $S_1$ and replays it; when transmitting bit "0," the sender randomly selects a packet interval from set $S_0$ and replays it.

However, the packet interval sent from set $S_0$ may be transferred to set $S_1$ due to the network jitter, causing a high bit error. Therefore, TRCTC must ignore the value of delay on part of the boundary to ensure that the receiver can correctly receive the secret messages.

L-N CTC (L-bits to $n$-packets scheme) proposed in [45] introduces a new data embedding method, which not only improves the channel capacity but also reduces the bit error rate. It can embed an L-bits secret message into the delays of N-consecutive packets. To represent different combinations of L-bits, the packet interval of L-N CTC will be evaluated around the normal network delay $d$ or exponential times of $d$.

*(2) Rate Modulation (C2).* The covert channel sender alters the data rate of a traffic flow from itself or a third party to the covert channel receiver. For example, Li et al. [46] analyzed a covert channel in the real switch. The sender exhausts the performance of a switch to affect the throughput of a connection from a third party to a receiver over time.

*(3) Package Arrival Time Modulation (C3).* This type of covert channel encodes hidden information by modifying the arrival time of multiple packets. For example, Tahir et al. [47] presented Sneak-Peek, a high speed covert channels in data center networks. In Sneak-Peek, the packets sent by the sender change some special packet sequences in the shared resource queue, and the receiver decodes secret messages from queuing delays of the special packets.

*(4) PDU Retransmission (C4).* In order to encode secret messages, the covert channel retransmits previously sent or received PDUs such as DNS requests, selected IEEE 802.11 packets, and selected TCP segments. Mazurczyk et al. [48] presented RSTEG (retransmission steganography). The main innovation of RSTEG is to not acknowledge a successfully received packet in order to force the sender to retransmit. The retransmitted packet will carry the secret messages.

*3.1.2. CSCs.* Abundant network protocols and characteristics make CSCs have multiple construction methods. We divide them into 5 categories: size modulation (C5), order modulation (C6), random value modification (C7), redundant field (C8), and multimedia data modification (C9).

*(1) Size Modulation (C5).* This type of covert channel encodes the covert information by changing the size of some special data such as the length of overall data packets and the length of a header element. The study in [49] discussed the technique of implementing CSCs by altering the size of TCP databursts. The TCP databurst is the number of TCP segments sent by a host before waiting for a TCP ACK packet.

*(2) Order Modulation (C6).* This type of covert channel encodes covert information by changing the order of PDU elements or header fields in the packet. The IPv6 extension header fields, header fields in the HTTP protocol, and options in the DHCP protocol are frequently used to encode covert messages [39].

*(3) Random Value Modification (C7).* If header elements in the network protocol contain random values, then these fields can be used to represent hidden information. The study in [17] proposed that the case and the least significant bit (LSB) of the values in some header fields can be used to encode secret messages.

*(4) Redundant Field (C8).* This kind of covert channel encoded hidden data into a reserved or unused header/PDU element. Rowland [50] proposed embedding covert channels in different unused areas in the IPv4 header and in the TCP header.

*(5) Multimedia Data Modification (C9).* This channel uses multimedia data (such as text, image, and video) as the carrier for secret transmission. For example, the information hiding is realized in [51] through embedding secret information in the characteristic data area of digital video by a steganography scheme based on chaotic mapping.

Wendzel et al. [39] categorize network covert channels at communication content level regarding three aspects (semantic, syntax, and noise). The semantic means whether the pattern modifies header elements in a way that leads to a different interpretation of the changed PDU. The syntax means whether the PDU structure is modified, and the noise means whether the channel is affected by noise. All CTCs do not change the structure of a PDU. But, they are greatly influenced by network delay or jitters, so they are all noisy and the robustness is not very well [38]. For CSCs, the fields used are not modified when transferring and hence there is few channel noise. But on the other hand, these channels are easy to be detected by the outside observer.

Besides the three aspects, CTCs have another characteristic, which is whether it is based on statistics [7]. For example, in many CTCs based on C1, in order to make the extra delay similar to the time series characteristics of the overt traffic, the delay will be carefully selected to ensure that it can not only fit the delay distribution of the overt traffic but also transmit secret messages.

So, we summarize the surveyed papers of CTCs and CSCs in Table 2 from the following aspects: category, semantic, syntax, noise, and statistics. In addition, we also give the description of each covert channel.

### 3.2. Construction Technology at the Transmission Network Level

*3.2.1. Proxy Technology.* Normally, users go directly to Internet sites to get network information, but nowadays, users often get information through the proxies. As a channel service in the Internet environment, the proxy service has multiple functions such as improving access performance, resource access control, and security protection and protecting user identity information. In addition, proxy services have the features of hidden, dynamic, and diverse. More importantly, proxies can help users break through content filtering restrictions and access the websites blocked by censors.

The proxy can be divided into two categories according to the traffic transmission path: E2E proxy (end-to-end proxy) and E2M proxy (end-to-middle proxy). Figure 2 shows the designs of E2E proxy (using HTTP proxy as one example) and E2M proxy (using Telex scheme as one example).

*(1) End-to-End Proxy*. The E2E proxy connects clients and servers directly. To get content from the server, the client sends a request to the proxy, and the proxy gets the content from the server and returns it to the client. The most common E2E proxy is the HTTP proxy [21], which acts like a web server and correctly accepts request and returns response. In order to extend the scope of the application protocol, the socks proxy is proposed. It simply passes data packets and does not care what application protocol they are. With the development of tunneling and cryptography, VPN proxy has emerged. It uses the technology of tunneling, encryption, decryption, and identity authentication, which means it is more secure. Existing mature VPN proxies include PPTP (Point-to-Point Tunneling Protocol) and L2TP (Layer-Two Tunneling Protocol), which are located in the second layer of the TCP/IP protocol, VTP (VLAN Trunking Protocol), and IPSec (IP Security), which are located in the third layer of the TCP/IP protocol.

However, E2E proxies are exposed to adversaries, and their activities are easily spotted. The global and active adversaries are able to block many of E2E proxies by discovering and banning the IP addresses of the servers on which they rely [20, 22]. To overcome this problem, researchers propose the E2M proxy.

*(2) End-to-Middle Proxy*. The traditional E2E proxy relays data to a specified server. Different from that, the E2M proxy is located in the path to a server and it can redirect the connection to an alternative destination. The E2M proxy needs a router at the friendly ISP to host it so that it can control the connection to an unblocked decoy server. Then, E2M proxy determines whether to block the connection and redirect it to a censored server by recognizing a steganographic tag. From the perspective of the censor, the E2M proxy user appears to be in contact only with the decoy server. The censor cannot block E2M proxy without blocking all connections that pass through participating ISPs, which is a large, primarily legitimate category of Internet traffic. So, E2M proxy will provide increased resistance to IP blocking.

There are four existing publications on end-to-middle proxy: Telex [22], Decoy Routing [63], Cirripede [64], and Tapdance [20]. The designs for the four systems are largely similar, although there are differences in some aspects, such as the embedded steganographic tag, blocking strategy, and deployment requirement. The comparison of the four E2M proxies is shown in Table 3.

*3.2.2. Anonymity Communication Technology.* The anonymous communication system is designed to access content blocked by censored anonymously. It uses technologies such as anonymous domain generation, traffic obfuscation, and broadcast/multicast to covertly forward messages. It can prevent attackers from acquiring communication relationships or the identity of the senders and receivers. That is, in the anonymous communication system, not only the channel is covert but also the identity of both parties in the communication.

The concept of mix-net proposed by Chaum [65] in 1981 is considered the origin of anonymity communication system. The core idea of mix-net is to encrypt and obfuscate messages based on mix nodes. Inspired by the mix-net, many anonymous communication systems are proposed, such as Crowds [66], P5 (Peer-to-Peer Personal Privacy Protocol) [67], Tor (the Second-Generation Onion Routing) [23], and I2P (Invisible Internet Project) [24]. Tor is currently the most active and most popular anonymous communication system. It has about 8 million daily users [68]. The core idea of Tor is onion routing [69]: firstly, select three suitable relay nodes and establish links with these relay nodes hop by hop; secondly, the client encrypts the secret messages 3 times; then, each relay node decrypts them in order. No node can know whether its previous node in the chain is the sender or the relay node. Likewise, no node can know whether its next node in the chain is the receiver or the relay node, so Tor can protect user privacy well.

According to different transmission delays, anonymous communication systems can be divided into high-latency systems and low-latency systems. High-latency systems are based on mix-net. High-latency systems are used for applications that can tolerate delays, such as anonymous e-mail services. However, most applications require timeliness, such as web browsing and live chat. Therefore, low-latency systems have a wider range of applications and attract more attention. Most of the research related to anonymous communication is also about low-latency systems. According to the network structure, the low-latency system can be divided into P2P anonymous network and non-P2P anonymous network. According to whether the routing path is determined, P2P anonymous network can be further divided into structured and unstructured network models. In a structured anonymous network, it is determined which nodes

Table 2: The summary of typical CSCs and CTCs.

| Covert channel | Category | Samanic preserving | Syntax preserving | Noiseless | Statistics | Description |
|---|---|---|---|---|---|---|
| Cabuk et al. [52] | C1 | √ | √ | × | × | The on-off CTC. |
| Shah et al. [43] | C1 | √ | √ | × | × | The jitterbug CTC. |
| Brodley and Spafford [44] | C1 | √ | √ | × | √ | The time-replay CTC. |
| Sellke et al. [45] | C1 | √ | √ | × | √ | The L-N CTC. |
| Liu et al. [53] | C1 | √ | √ | × | √ | A improved method for selecting interval time in [44]. |
| Li et al. [46] | C2 | √ | √ | × | × | Consuming switch performance to affect throughput. |
| Tahir et al. [47] | C3 | √ | √ | × | × | A high speed covert channels in data center networks |
| Zhang et al. [54] | C3 | √ | √ | × | × | Modify numbers of video packets. |
| Ahsan and Kundur [55] | C3 | × | √ | × | × | Modify the order of IPSec packets. |
| Zhang et al. [56] | C3 | √ | √ | × | × | Postpone or extend the silence periods over VoLTE. |
| Krtzer et al. [57] | C4 | √ | √ | × | × | Duplicate selected IEEE 802.11 packets. |
| Mazurczyk et al. [48] | C4 | √ | √ | × | × | Retransmit a packet which carries secrets. |
| Schulz et al. [58] | C5 | √ | × | √ | — | Modulate the size of IPSec packets. |
| Luo et al. [49] | C5 | √ | × | √ | — | Modulate the size of TCP databursts. |
| Rios et al. [59] | C6 | × | × | √ | — | Adjust the options in the DHCP protocol. |
| Zhang et al. [60] | C6 | × | × | √ | — | An enlarging-the-capacity packet sorting covert channel. |
| Wang et al. [17] | C7 | × | × | √ | — | Utilize LSB in some header fields. |
| Trabelsi et al. [61] | C7 | √ | × | √ | — | Utilize ICMP payload. |
| Rowland [50] | C8 | × | × | √ | — | Utilize unused areas in the IPv4 and TCP header. |
| Lucena et al. [31] | C8 | × | × | √ | — | Utilize unused areas in the IPv6 header and its extensions. |
| Liu et al. [51] | C9 | √ | × | √ | — | A video steganography scheme based on chaotic mapping. |
| Kadhim et al. [62] | C9 | √ | × | √ | — | A image steganography scheme based on the mapping function of genetic algorithm. |

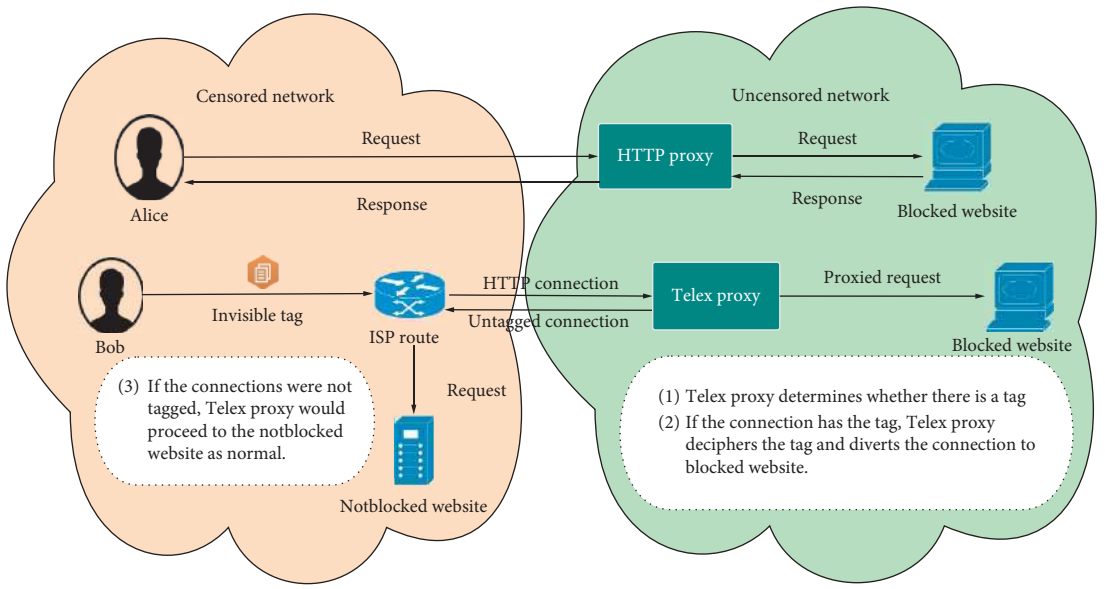


Figure 2: E2E and E2M concept (example users). Alice is connecting to the HTTP proxy, and Bob is connecting to Telex proxy.

Table 3: E2M proxy.

| E2M | Tag composition | Tag location | Blocking strategy | Deployment requirement |
|---|---|---|---|---|
| Telex [22] | (i) An ECDH public key point. (ii) A hash of the ECDH secret shared with ISP. | TLS client nonce | Only tagged flow | Inline-blocking and redirecting components provided by ISP. |
| Cirripede [64] | (i) An ECDH public key point. (ii) A hash of the ECDH secret shared with ISP. | TCP ISNs | All connections | Inline-blocking and redirecting components provided by ISP. |
| Decoy routing [63] | (i) An HMAC of the previously established shared secret key. (ii) The current hour. (iii) A per-hour sequence number. | TLS client nonce | Only the tagged flow | Inline-blocking and redirecting components provided by ISP. |
| Tapdance [20] | The client's connection-specific elliptic curve public key point. | TLS ciphertext | Not blocking | (i) A passive tap that observes traffic transiting the ISP. (ii) The ability to inject new packets. |

data stream will pass through, while in an unstructured anonymous network, the path of the data stream is unknown.

Besides the mix-net, anonymous communication systems also use many techniques to increase the system's covertness. The common techniques include anonymous domain generation, broadcast/multicast, probabilistic traffic routing mechanism, and traffic obfuscation. Some newer technologies are also used, such as zero knowledge proofs in [8] and verifiable shuffle technique in [70]. But because they are specific to a certain system, we will not go into details.

The anonymous domain generation is a mechanism similar to DGA (domain generation algorithm) used by malicious software. It uses a random private key, calculates the corresponding public key, and uses the public key as part of the domain. So, it can guarantee the anonymous communication system's anonymity and security.

The core idea of broadcast/multicast technology is each node broadcasts a message to other nodes in the system in each cycle of the system operation. DC-Nets (Dining Cryptographers) [71] is a typical example of using this technology. It is an anonymous communication system based purely on broadcast/multicast. An improved example is P5 [67], which divides the nodes into multiple broadcast groups to improve the scalability of the system.

The probabilistic traffic routing mechanism is often used to build unstructured anonymous networks. This mechanism is implemented through DHT-based (distributed hash table) routing protocols and random walk protocols. In systems that use probabilistic traffic routing mechanism, nodes can decide whether to forward traffic to the next node based on a certain probability. Because it is difficult for each node on the path to determine whether its predecessor is the original sender of the message or just an intermediate forwarding node, it effectively guarantees the anonymity of the original sender. Crowds [66] is a typical example of using this mechanism. Other examples are Torsk [72] and BitBlender [73].

Traffic obfuscation can erase or randomize the statistical characteristics of covert traffic, so that the load of covert traffic looks like a uniform random bitstream or a "benign" protocol [74]. Its goal is to make it difficult for adversaries to distinguish between the obfuscated traffic and overt traffic. Current traffic obfuscation techniques are as follows:

(i) Randomization: randomization refers to the use of encryption, random padding, and other methods to randomize the characteristics of covert traffic. For example, the Tor project [75] has developed a variety of randomization mechanisms, including Obfsproxy3 [76], Obfsproxy4 [77], Dust [78], and ScrambleSuit [79].

(ii) Protocol mimicry: the main idea of protocol mimicry is imitating or masquerading as popular whitelisted protocols which are rarely suspected by adversaries. For example, SkypeMorch [80] is a transport layer plugin that integrates traffic between Tor clients and Tor bridges into Skype traffic. Another example is StegoTorus [81]. The core idea is to segment Tor traffic and simulate other overt traffic such HTTP.

(iii) Join dummy traffic: to provide stronger anonymity, some systems generate additionally dummy traffic. This technology is used in many existing system, such as P5 [67] and Loopix [11].

(iv) Tunneling: the tunneling technology is one extreme of the mimicry logic, which simply encapsulated data into an (usually encrypted) overlay protocol. A famous example is meek [82] deployed with Tor. Meek core idea is to use different domain names in different places, one in the SNI for DNS requests (set to noncensored URL) and the other in the HTTP host field (set to censored URL). Meek uses cloud platform as relay node and redirects Tor traffic to Meek server.

We summarize the main technologies used in each anonymous communication system and application scenarios in Table 4.

## 4. Covert Channels in the New Network Environment

*4.1. Covert Channel Based on Streaming Media.* With the popularity of video and audio services such as YouTube and Skype, the audio and video are becoming the main business type in mobile networks. The audio and video traffic accounts for the vast majority of the entire Internet traffic.

Table 4: The anonymous communication systems.

| System | Low latency | P2P network | Structured network | Main techniques | Application scenarios |
|---|---|---|---|---|---|
| Mix [65] | × | — | — | (i) Mix-net | (i) e-mail |
| DC-Nets [71] | × | — | — | (i) Broadcast/multicast | (i) Anonymously post messages |
| Tor [23] | √ | √ | √ | (i) Onion routing<br>(ii) Tunneling tool: meek<br>(iii) Protocol mimicry tool: StegoTorus and SkypeMorch<br>(iv) Randomization tool: Obfs, dust etc<br>(v) Anonymous domain generation | (i) Anonymously web browsing<br>(ii) Live chat |
| P5 [67] | √ | √ | √ | (i) Mix-net<br>(ii) Join dummy traffic<br>(iii) Broadcast/multicast | (i) Anonymous web transactions<br>(ii) Anonymous re-mailers |
| Crowds [66] | √ | √ | × | (i) Mix-net<br>(ii) Probabilistic traffic routing mechanism | (i) Anonymously web browsing |
| Torsk [72] | √ | √ | × | (i) Mix-net<br>(ii) DHT-based routing protocols | (i) Anonymously web browsing<br>(ii) Live chat |
| BitBlender [73] | √ | √ | × | (i) Mix-net<br>(ii) DHT-based routing protocols | (i) Bitcoin mixer |
| Anonymizer [83] | × | — | — | (i) Mix-net | (i) Anonymously web browsing<br>(ii) Anonymous e-mail services |
| Mixminion [84] | × | — | — | (i) Mix-net<br>(ii) Anonymous domain generation | (i) Anonymous e-mail services |
| Babel [83] | × | — | — | (i) Mix-net | (i) Anonymous e-mail services |
| I2P [24] | √ | √ | √ | (i) Garlic routing (a variant of onion routing)<br>(ii) Anonymous domain generation | (i) Anonymously web browsing<br>(ii) File transfer<br>(iii) Instant messaging |
| Atom [8] | √ | √ | × | (i) Mix-net<br>(ii) Zero knowledge proofs | (i) Communication bootstrapping<br>(ii) Microblogging application |
| Riffle [70] | √ | √ | √ | (i) Onion routing<br>(ii) Verifiable shuffle technique | (i) File sharing<br>(ii) Microblogging applications |
| Loopix [11] | √ | √ | √ | (i) Mix-net<br>(ii) Join dummy traffic (loop traffic created by users and mix servers) | (i) Private e-mail<br>(ii) Instant messaging |

According to the Mobile Network Visualization Network Index (VNI) Forecast Report (2017–2022) released by Cisco (available under https://www.cisco.com/c/en/us/solutions/collateral/serviceprovider/visual-networking-index-vni/white-paper-c11-738429.html), IP video traffic will be 82 percent of all IP traffic (both business and consumer) by 2022, up from 75 percent in 2017. And, the Internet video traffic will grow fourfold from 2017 to 2022, a CAGR (Compound Annual Growth Rate) of 33 percent.

In this context, many covert channels based on streaming media have been proposed. They use audio and video traffic allowed by the observer as overlay traffic and build covert channels. The carrier is a popular encrypted streaming application such as Skype. This technique can help users watch the censored video, and it can work without requiring changes to the carrier application. There are three systems that have implemented this technique: Facet [25], CovertCast [26], and DeltaShaper [27].

Facet [25] is a covert communication system for transmitting censored video, and it relies on the assumption that the observer is unwilling to indiscriminately block all or most sessions of the cover protocol (Skype). To the outside observer, the Facet client is just having a Skype session. Facet consists of clients, Facet servers, and emulators. The procedure of a Facet connection is as follows:

(1) The Facet client and Facet server establish initial connections

(2) A Facet client sends a uniform resource locator (URL) of the censored video to the Facet server

(3) Facet server downloads video from blocked video sites such as YouTube, Vine, or Vimeo

(4) The emulator simulates the video content as a Skype session and resends it to the client at a lower resolution

(5) The Facet client ends the connection and the Facet server destructs the emulators and ends the session

Different from Facet, CovertCast [26] supports that multiple clients receive data transmitted in a specific live stream in the real time. And, CovertCast is scalable, with the server workload independent of the number of clients receiving content. To the observer, CovertCast traffic is similar to the traffic that users watch someone broadcasting on a given live-streaming platform such as YouTube. CovertCast consists of users, CovertCast clients, and CovertCast servers. The communication process of CovertCast is as follows:

(1) CovertCast server crawls a censored website and modulates its content into images.

(2) CovertCast server broadcasts images by live-streaming video services and begins to download the next website.

(3) CovertCast client constantly monitors the stream for new images. When it detects one, it demodulates the image and saves the extracted content.

(4) The user's web browser sends a request through the user's proxy.

(5) CovertCast client creates a response with the corresponding website.

In Facet or CovertCast, the format of covert messages is restricted to video (in Facet) or Web content (in CovertCast). In DeltaShaper [27], the covert TCP/IP packets are encoded and embedded into the video stream transmitted by the video channel of a popular videoconferencing application such as Skype between the communication endpoints, which means that DeltaShaper allows for tunneling arbitrary TCP/IP traffic. To the observer, the client and server are just engaged in a Skype session. DeltaShaper consists of client endpoint and server endpoint. The same procedure is applied at both endpoints of a Skype call, thus DeltaShaper supports bidirectional communication. The procedure is as follows:

(1) The sender modulates covert data into images and encodes them in a video stream which is fed to Skype

(2) Skype transmits this video to the receiver's Skype instance

(3) The receiver captures the stream from the Skype video buffer

(4) The receiver's decoder extracts the payload from the stream

We give an overview about the three covert channels based on streaming media with respect to the covert message transmitted platform, whether it is bidirectional and whether it supports multiple clients in Table 5.

*4.2. Building Covert Channels Using Blockchain Technology.* With the development of covert communication countermeasure technology, the traditional network covert channels based on TCP/IP architecture have the risk that the channel is regulated, the traffic is easy to be tracked, and the identity information of the communicators is easy to be recognized [85], which makes it difficult to meet the security requirements of data covert transmission. Blockchain is one of the representatives of the new generation of information technology. It has a large number of active users, a large number of transaction data packets, and many ways to embed the secret data. For example, there are many fields in the blockchain ledger structure that can store data. The blockchain network adopts transaction transmission mechanism based on flood forwarding, which ensures that information can be effectively, quickly, and reliably transmitted to all nodes in the network. A decentralized flooding mechanism can also protect the recipient of a covert transmission of data by avoiding stealing communication privacy by monitoring a single server or a single communication link.

Li et al. [28] proposed a model of covert timing channel in the blockchain network and uses a formal method to model and proves the anti-interference and tamper-resistance. Secondly, they constructed a scenario of the covert channel in the blockchain network based on the time interval of business operations. They also present covert channel evaluation vectors for blockchain networks containing detection resistance, robustness, and transmission efficiency.

Brenner et al. [10] proposed a model of covert storage channel in the blockchain. They explored the possibility of applying blockchain technology to the transmission of C&C (command and control) instructions in a field of blockchain protocol and described the prototypes of Zombiecoin [10], which are based on Bitcoin.

Besides the researches on the model of blockchain-based covert channel, there are three systems that have been actually deployed: Zombiecoin [86], Botchain [29], and Chaninchannels [30]. The designs of these three systems are very similar. In the three systems, communication participants are expected to covertly transmit messages through blockchain. Firstly, they apply to be the client nodes of the blockchain network and negotiate labels in advance so that the receiver can identify the transactions containing covert messages from thousands of transactions. To ensure security, both sides also need to negotiate the encoding, encryption algorithm, and the way of message embedding. Then, the sender encodes, encrypts, and embeds the messages into certain transactions according to the negotiated algorithm and sends them to the server nodes of blockchain. After the flooding propagation mechanism of blockchain network, the receiver identifies the special transactions through negotiated labels and extracts the covert messages.

Ali et al. [86] proposed Zombiecoin 2.0, which validated the claims in [10] and deployed successfully over the

Table 5: Streaming media-based covert communication systems.

| | Facet [25] | CovertCast [26] | DeltaShaper [27] |
|---|---|---|---|
| Covert message | Video | Web content | TCP/IP traffic |
| Platform | Skype | YouTube | Skype |
| Bidirectional | × | × | √ |
| Multiple clients | × | √ | × |

blockchain network. In this system, the covert messages are directly inserted in the output script function OP_RETURN (available under https://en.bitcoin.it/wiki/OP_RETURN), which is a field of particular blockchain implementation and originally used to carry additional transaction information. The sender and the receiver negotiate a pair of prenegotiated public-private keys as the label to identify the transactions that contain covert messages. The receiver identifies these transactions by scanning the ScriptSig (the unlocking script in Bitcoin to verify whether a transaction is passed) which contains the sender's public key and the digital signature (computed over the transaction) using the private key. The receiver verifies the signature and decodes the messages.

Chainchannels [30] realized a new way of embedding covert messages in blockchain with key leakage and did some cryptographic proofs. It uses a subliminal channel in digital signatures to insert secret messages totally. Many blockchain-based virtual currencies use the ECDSA (elliptic curve digital signature algorithm,) and the subliminal channel can substitute the nonce used in ECDSA with the secret messages. The cryptographic characteristics of ECDSA ensure that no errors will occur during the process of extracting the secret message, thus increasing the system. In terms of the label to identify the special transactions, Chainchannel uses a pair of prenegotiated public-private keys as the label, which is the same as Zombiecoin 2.0.

Botchain [29], proposed by Cybaze, is a fully functional botnet which is based on Bitcoin protocol. This system also utilizes OP_RETURN to embedding secret messages, which is similar to Zombiecoin 2.0 [10]. In terms of the label to identify the special transactions, Botchain [29] uses prenegotiated virtual currency wallet addresses as the label, which is different from the Zombiecoin 2.0 and Chainchannels.

We compare the three systems in term of five aspects: secret message embedding method, used field to embed secret messages, label embedding method, used field to embed labels, and suitable platform in Table 6.

*4.3. IPv6 Covert Channels.* IPv6 (Internet Protocol version 6, also called the NextGeneration Internet Protocol or IPng) is a next-generation IP protocol designed by the Internet Engineering Task Force (IETF) to replace IPv4. The IPv6 header field is reduced by four (header length, identification, flags, fragment offset, and header checksum), and the options are replaced with extended headers.

However, the grammar rules of the IPv6 are not perfect, which makes the construction of IPv6 covert channels very easy. In [31], several possible covert channels have been analyzed in the IPv6 header and its extensions. In [87], Yang think that IPv6 packets are a good carrier for information hiding and propose potential covert channels. Ullrich et al. [88] also discuss the use of IPv6 covert channels.

We conclude that there are four types of IPv6 covert channels. The first type is based on reserved fields of the IPv6 extended headers. The fields that can be used are as follows:

(i) The router alert options in the hop-by-hop option header (2 bytes/packet)

(ii) The reserved field in the routing extension header when routing type is 0 (4 bytes/packet)

(iii) The reserved field in the fragment extension header (10 bits totally/packet)

(iv) The reserved field in the authentication extension header (2 bytes/packet)

(v) The binding update option in the destination options header (4 bits/packet)

The second type is based on the order. Covert channels of this type hide information by ordering several special parameters. This method does not insert extra characters, so the risk of channel exposure is relatively low.

(i) Encoding covert messages based on the difference between the order of extension headers and the suggested order of RFC2460 (available under https://tools.ietf.org/html/rfc2460) (8 bits/packet)

(ii) Encoding covert messages based on the difference between the order of $N$ addresses in 0 routing type header and preshared order ($N$ bits/packet)

The third type is based on some random values. The reason for this type of channel is that some fields are incompletely defined or the design of the inspection mechanism is not strict. For example, when the reassembly process is performed, the destination host only inspects the next header value of the first fragment and ignores the next header values of fragments that differ. This causes that the sender can set a false next header value to transmit secret messages:

(i) Set one or more false router addresses in the routing extension header when routing type is 0 (up to 2048 bytes/packet)

(ii) Set false values in the field of traffic class, flow label, hop limit, and source address in the IPv6 header (8 bits, 20 bits, 1 bit, and 16 bytes/packet, respectively)

(iii) Set a false padding value in the extension headers including hop-by-hop option extension header, destination option extension header, and ESP extension header (up to 256 bytes/packet)

(iv) Set a false next header in the IPv6 header (varies) and fragment extension header (at least 8 bits/fragment)

The channels proposed above are all covert storage channels. The three types of channels can be classified in turn into C8, C6, and C7 introduced in Section 3. The fourth type is based on tunneled traffic. Tunneling technology is not only used in anonymous communication systems but also used to

Table 6: Blockchain-based covert communication systems.

| | Zombiecoin [10] | Botchain [29] | Chainchannels [30] |
|---|---|---|---|
| Message embedding method | Directly embedded | Directly embedded | Subliminal channel |
| The field to embed messages | OP_RETURN | OP_RETURN | Digital signature |
| Label embedding method | Directly embedded | Directly embedded | Directly embedded |
| The field to embed labels | Public key | Wallet addresses | Public key |
| Platform[†] | Bitcoin only | Bitcoin only | All blockchain networks |

[†]The virtual currency the system is suitable for.

enable IPv6 packets to penetrate the IPv4 network, which results in the existence of IPv6 covert channels in tunneled traffic. The sender can embed secret messages into a IPv6 tunnel packet. The tunnel technology used in IPv6 network including ISATAP [89], 6to4 [90], and 6over4 [91].

*4.4. Summary.* The characteristics of the new network bring natural convenience to the construction of network covert channels. For the covert channel based on streaming media, the real-time and interactive features improve the transmission efficiency of secret messages. In addition, the widespread popularity of streaming media applications has also made channels more difficult to expose. For the covert channel based on blockchain, the participant-anonymity can protect user identity, and flooding propagation makes it impossible for observers to determine the true recipient of the secret messages and to know who is receiving the messages. In addition, the tampering resistance strongly guarantees the robustness. For the IPv6 covert channel, the extended headers and grammar rules open up multiple possible construction methods.

On the other hand, due to the complexity of the new network environment, there are few effective attacks against the three new type covert channels (the attacks include elimination, limitation, and detection, which are introduced in Section 6). Therefore, compared with the traditional network covert channels, they are more secure.

## 5. Network Covert Channel Metrics

The evaluation metrics of the network covert channel include 3 aspects: concealment, robustness, and transmission efficiency. Concealment refers to the ability to be undetected by adversaries. In this regard, we propose for the first time to divide the metrics of concealment into message concealment and identity concealment. Robustness refers to the ability of the network covert channel to accurately transmit data. And, transmission efficiency refers to the maximum rate at which the channel can transmit data without error.

*5.1. Concealment.* A successful network covert channel requires high concealment. It includes not only the concealment of the communication content but also the concealment of the identity of the communicating parties. The adversaries can expose the covert channel through these two aspects. So, we list the concealment metrics as follows.

*5.1.1. Concealment of Communicating Parties.* The concealment of communicating parties consists of anonymity and unobservability. Anonymity means that users can communicate without disclosing their identity. Unobservability refers to the indistinguishable state of traffic with covert messages in the overt traffic set.

*(1) Anonymity*. There are many methods proposed to measure the anonymity [92–94], which can be categorized into three classes: measurement based on continuous interval, measurement based on the size of anonymous set, and measurement based on entropy.

(1) Measurement based on continuous interval: Reiter and Rubin [92] described the degree of anonymity, which is widely adopted. The degree of anonymity is defined as a continuous interval which ranges from absolute privacy to provably exposed. The six key points are as follows:
   (i) Absolute privacy: the adversary cannot even perceive the presence of communication
   (ii) Beyond suspicion: though the attacker can see evidence of a sent message, the sender appears no more likely to be the originator of that message than any other potential sender in the system
   (iii) Probable innocence: the observer thinks that each sender appears no more likely to be the originator than to not be the originator
   (iv) Possible innocence: the originator of the message is likely to be someone else
   (v) Exposed: the originator of the message is unlikely to be someone else
   (vi) Provably exposed: the adversary can prove the identity of sender or receiver to others

(2) Measurement based on the size of anonymity set: Berthold et al. [93] proposed the degree of anonymity can be defined by the size of the group. For example, the anonymity may be measured as follows:

$$A = \log_2(N), \tag{1}$$

where $N$ is the number of possible senders.

(3) Measurement based on entropy: Diaz et al. [94] issued that because an observer may have some background knowledge, objects in an anonymous set may have different probabilities. So, they proposed

the measurement based on entropy. The entropy of sender set is defined as equation (2), where $S = \{s_1, s_2, \ldots, s_N\}$ is the anonymity set and $p_i$ is the probability that the possible object $s_i$ is the real sender. The $p_i$ is concluded by the observer's background knowledge:

$$H(X) = -\sum_{x=i}^{n} p_i \log_2(p_i). \tag{2}$$

So, $A - H(X)$ represents the information obtained by an observer who has known some background knowledge. The degree of anonymity $d_a$ is defined as

$$d_a = 1 - \frac{A - H(X)}{A}. \tag{3}$$

*(2) Unobservability*. Anonymity can measure the security of the identity information of the communication subject, while unobservability is a measure of the user's behavioral security. Adversaries with traffic analysis capabilities can use protocol fingerprint characteristics to determine the user behavior, such as whether the senders or receivers are sending or receiving messages.

Tan et al. [18] proposed to use the relative entropy $D$ between the communication behavior of the covert communication system and the overt network behavior to measure the unobservability. The degree of unobservability $d_u$ is defined as equation (4). The $D_m$ is the max. relative entropy:

$$d_u = 1 - \frac{D_m - D}{D_m}. \tag{4}$$

When the probability distribution of the network behavior seen by the attacker is completely consistent with that of the normal network behavior, the relative entropy between them can be minimized, that is, the unobservable degree $d_u$ is minimized.

*5.1.2. Concealment of Communication Content.* Since constructing a covert channel based on time information modulation will cause some time characteristics of the channel to change, most of the research on the metrics of message concealment is aimed at the CTCs. Due to the technical specificity of other types of covert channel, there is no universal method to measure the concealment of these channels [7]. So, we list the following methods used to measure the message concealment of CTCs.

*(1) Kolmogorov-Smirnov Test*. KS test (Kolmogorov-Smirnov test) points an upper bound between the cumulative probability of experience and the cumulative probability of the target distribution at each data point [95]. Archibald and Ghosal [96] leverage it as a method to assess the concealment of covert channel. How to calculate the test score is shown as

$$D_n = \max_x \{|F(x) - G(x)|\}, \tag{5}$$

where $F(x)$ and $G(x)$ are empirical cumulative probability distributions of the IPDs of the overt traffic and covert traffic, respectively.

*(2) Kullback-Leibler Divergence Test*. The KL divergence is a measure of relative entropy between two target distributions [97], which means the KL divergence is used to show the distance between two random variables. It is leveraged as a metric for detecting CovertCast [98] and CTCs in [96, 99]. The KL divergence from $P$ to $G$ is denoted as

$$D_{KL}(P \parallel G) = \sum_x p(x) \cdot \log\left(\frac{p(x)}{g(x)}\right). \tag{6}$$

The $p(x)$ and $g(x)$ are two probability distributions of the covert traffic sample and the overt traffic sample, respectively.

*(3) Standard Deviation Test*. Wu et al. [100] use the dispersion of standard deviation to assess concealment. It can measure the variation of a stream throughout the transmission process. This method starts by separating traffic into nonoverlapping windows of size $w$. Then, the standard deviation is computed for each window, which is shown in equation (7). Finally, as presented in equation (8), it calculates the standard deviation of the pairwise difference as the metric of concealment $C_t$:

$$\sigma = \mathrm{STDEV}(X) = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \overline{x})^2}{n}}, \tag{7}$$

$$C_t = \mathrm{STDEV}\left(\frac{\left|\sigma_i - \sigma_j\right|}{\sigma_i}\right), \quad i < j,\ \forall i, j. \tag{8}$$

*(4) Entropy-Based Test.* Because covert channels will cause the change of ER (entropy rate), it is used to detect covert channels in [28, 85, 101, 102]. The ER describes the uncertainty of a random variable sequence, and the sequence length $m$ approaches infinity. It is defined as

$$H(\mathcal{X}) = \lim_{m \longrightarrow \infty} \frac{H(X_1, X_2, \ldots, X_m)}{m}. \tag{9}$$

Porta et al. [103] proposed to use the entropy rate of finite samples as the estimated value of ER by calculating CCE (corrected conditional entropy), which is shown in equations (10) and (11). The estimated value of ER is the minimum value of CCE when $m$ changes:

$$\mathrm{CCE}\left(X_m \mid X_{m-1}\right) = H\left(X_m \mid X_{m-1}\right) + p(X_m) H(X_1), \tag{10}$$

$$\overline{\mathrm{ER}} = \min_{i=1,m}\left(\mathrm{CCE}\left(X_i \mid X_{i-1}\right)\right). \tag{11}$$

*5.2. Robustness.* Many studies use BER (bit error rate) to measure the robustness of the network covert channel. The

lower the BER, the higher the robustness. There are currently two definitions of BER:

(i) The traffic containing covert messages may be affected by network noise or adversary noise, resulting in errors in the messages received by the receiver. So, the first definition [28] is the error probability obtained by comparing the number of error bits $S_{\text{error}}$ to the total number of bits transmitted $S_{\text{all}}$. The BER is defined as

$$\text{BER} = \frac{S_{\text{error}}}{S_{\text{all}}}. \tag{12}$$

(ii) The other aspect to define BER is from the perspective of message recovery. Because the encoding method used by the covert channel may lose some of the original information, there will be errors in the process of decoding the final message after the receiver obtains the covert message. Houmansadr and Borisov [104] define the BER as the error probability obtained by comparing the original message to the final message after decoding. The definition is shown in equation (13), where $e(x, y) = 1$ for $x \neq y$ and $e(x, y) = 0$ for $x = y$:

$$\text{BER} = \frac{\sum_{i=1}^{k} e\left(m(i), m'(i)\right)}{k}, \tag{13}$$

where $k$ is the decoded message length; $m(i)$ is the $i$-th bit of the original message; $m'(i)$ the $i$-th bit of the message obtained after transmission.

*5.3. Transmission Efficiency.* The transmission efficiency of the network covert channel is evaluated using the amount of information contained in a unit symbol or the amount of information transmitted in a unit time. There are three definitions about the transmission rate, which are enumerated as follows:

(i) Wu et al. [100] define the transmission efficiency as the maximum possible error-free information transmission rate. Equation (14) shows the definition. The $N(t)$ represents the amount of information transmitted by $N$-ary coding in time $t$:

$$C = \frac{N(t)}{t}. \tag{14}$$

(ii) Houmansadr and Borisov [104] proposed to take each covert data packet as a unit. They define the transmission efficiency as the number of bits of covert messages transmitted by each data packet. The definition is shown as follows:

$$r = \lim_{N \longrightarrow \infty} \frac{K}{N}, \tag{15}$$

where $K$ is the number of bits of covert messages sent using $N + 1$ packets.

(iii) Li et al. [28] discussed the transmission efficiency in blockchain network environment. It is defined as the amount of information transmitted per unit time. The following equation shows the transmission efficiency $C$:

$$C = va \sum_{i=1}^{S} p_i 1bp_i, \tag{16}$$

where $v$ is the information carrier transmission rate; $a$ is the number of modulation symbols that each covert information carrier can carry; and $p_i$ is the probability of occurrence of the $i$-th encoded character in the encoding table.

(iv) Wang et al. [105] use channel capacity as a method to measure the transmission efficiency of CTCs. It is defined as

$$\text{capacity} = \frac{\text{bit}}{\overline{\text{ipd}}}, \tag{17}$$

where bit is the amount of information carried by each packet interval, and $\overline{\text{ipd}}$ is the average network packet interval.

## 6. Attacks against Network Covert Channels

Attacks against network covert channels can be divided into three categories according to [35]:

(i) Elimination: removing covert channels or making the covert channel completely unusable

(ii) Limitation: reducing the transmission efficiency of covert channels

(iii) Detection: discovering the existence of covert channels or the identity of both parties in the communication

Eliminating attacks is the most difficult to perform. On the one hand, attackers need the ability to monitor and modify traffic, such as national censors. On the other hand, the specificity of certain channel structures makes it impossible to eliminate them fundamentally. For the limitation attack, while it works, it may also interfere with normal network communication. So, most researches on attacks have focused on detection. In the remainder of this section, we will give attacks against different network covert channels.

### 6.1. Attacks against Content Level Channels

*6.1.1. CSCs.* For CSCs, there is no effective way to carry out limitation attacks. But the traffic normalization (TN) method can effectively eliminate CSCs. TN can standardize fields in various protocols, so these fields cannot be arbitrarily filled with additional information. This makes most CSCs unavailable. In addition, most of the current detection attacks for CSCs are based on analyzing traffic fingerprint. Traffic fingerprint is a feature or a series of feature

combinations that can represent certain traffic, such as packet length and ISN sequence. Attackers train the ML or DL classifier based on collected traffic fingerprints for normal communication behavior and use the classifier to detect CSCs.

*6.1.2. CTCs.* For CTCs, since time-dependent features are difficult to regularize, using TN to eliminate CTCs is not realistic. However, by adding delays to the covert channel, it can greatly affect the transmission efficiency of CTCs. In addition, detection attacks against CTCs have been studied in recent years and can be divided into two categories. One is to use statistical methods to detect the shape, regularity, and randomness of traffic. The other is to use ML or DL technology like the detection attacks against CSCs, except that the fingerprints used are time-dependent.

*6.2. Attacks against Network Level Channels.* Due to the particularity of the network level covert channel construction methods, no research has shown that they can be effectively limited. In the following, we will introduce the attacks on the network level channels from the aspects of elimination and detection.

*6.2.1. Proxy.* Circumventing Internet censorship is a mechanism commonly used by censors. It contains IP blocking, URL blocking, DNS hijacking, keyword filtering, network protocol blacklist/whitelist, etc. Among them, the IP blocking and URL blocking can make proxy server unusable and eliminate it. In addition, there are three methods for detection attacks against proxy:

(i) The attacker first extracts the characteristics of the packet and generates regular expressions based on this. And then, the attacker inspects the content of the traffic based on regular expressions.

(ii) The attacker injects the traffic watermark into traffic and observes if the traffic from the target host contains the watermark.

(iii) The attacker uses ML or DL technology to find the traffic produced by proxy. The key to this approach is to determine the traffic fingerprint of the corresponding proxy service.

*6.2.2. Anonymous Communication System.* Circumventing Internet censorship is also used to eliminate this channel. For example, censors block the IPs of some known entry nodes and bridge nodes in Tor [33, 80]. Likewise, the research [34] measured I2P censorship at a global scale and found that censors can hinder access to I2P using several blocking techniques, such as URL blocking and DNS hijacking.

Detection attacks on anonymous communication systems include two aspects. One is to detect covert communication traffic to discover network nodes. For example, He et al. [106] propose Tor traffic could be identified using TLS fingerprint (cipher suite and digital certificate) or message length distribution characteristics. Wang et al. [85] perform detection attacks against obfuscation tools which are configured in Tor. The other is to detect the association between nodes, in order to discover the connection between sender and receiver, which destroys the anonymity of the channel. This attack includes the following ways:

(i) Predecessor attack: the adversary has control some nodes and collects relevant information. When he knows the node is on the senders' path, the precursor node of this node is more likely to be a sender. This attack requires many controlled nodes to work together, so the predecessor attack is also called the collusion attack.

(ii) Sybil attack: Sybil attack means that malicious attackers control some nodes by imitating the identity of nodes in the system. These malicious nodes leak system information to the attacker, and the attacker can infer the routing forwarding and data redundancy strategies of the system, so as to launch a precursor attack.

(iii) Replay attack: replay attack means that an attacker records the message to be tracked first and then sends it back. The attacker tracks the message by observing the output of the mix node until the recipient is found. In deterministic encryption schemes, resisting replay attacks is a difficult problem. Mix nodes must remember the message they have processed in order to prevent attackers from discovering recipient.

(iv) Message tagging attack: tagging attack is initiated by an internal attacker who controls the first and last node. The attackers mark messages at the first node. In this way, the attacker can identify the message in the last node according to the tag, thus linking the sender and receiver.

(v) $N$-1 attack: the $N$-1 attack is also known as flooding attack. The attacker's goal is to track the path of a target message. He isolates any message other than a target message, and a certain number of forged messages are sent at the same time. Thus, when all messages flow out of mix, the only message that is not forged is the target message that the attacker wants to track.

(vi) Flow correlation attack: the adversary observes the traffic that one particular mix is receiving at a special port and then finds the corresponding traffic at output ports. This attack can be performed by noting the timing of the packet between the traffic at the input and the output port or with the help of ML and DL to correlate traffic.

*6.3. Summary.* Due to the existence of a large number of network covert channel construction techniques, each attack method is generally only effective for specific channels. Even the elimination or limitation attacks against some channels have not been studied. Now, research focuses on detection

Table 7: Attack against covert channels.

| Level | Covert channel | Elimination | Limitation | Detection |
|---|---|---|---|---|
| Communication content | Covert timing channels | — | Adding delays | (i) Detecting traffic shape.<br>(ii) Detecting traffic regularity.<br>(iii) Detecting traffic randomness.<br>(iv) ML or DL technology. |
| | Covert storage channels | Traffic normalization | — | ML or DL technology. |
| Transmission network | Proxy | (i) IP blocking<br>(ii) URL blocking | — | (i) Regular expressions.<br>(ii) Traffic watermark.<br>(iii) ML or DL technology. |
| | Anonymous communication | (i) Blocking the IPs of mix nodes<br>(ii) Blocking official homepages<br>(iii) Poisoning DNS resolutions | — | (i) Detecting mix nodes: ML or DL technology.<br>(ii) Detecting the association between nodes: Sybil attack, etc. |

attacks. Most detection attacks are done by collecting traffic fingerprints and training ML or DL models. The success of this attack lies in selecting the appropriate traffic fingerprint and model. We summarise the attacks against covert channels in Table 7.

## 7. Challenges and Future Directions

After investigating the attacks against covert channels in Section 6, we see that there are two challenges that still remain:

(i) The IP blocking or other blocking technology has made most transmission network level channels unavailable

(ii) The use of ML and DL technology makes the covert channel easier to expose

So, the future directions for covert channel are to improve the ability to resist attacks. In addition, we discuss several specific research methods in the following.

*7.1. Adversarial Examples.* In recent years, researchers have found that ML or DL shows great vulnerability when inputting some well-designed examples. These well-designed examples are adversarial examples. With the help of adversarial machine learning technology, we can add a certain amount of noise to the covert traffic to generate adversarial examples. So, the model is deceived to make a wrong judgment.

*7.2. Measuring Internet Censorship.* In order to avoid being eliminated by censors, the covert channel can be designed specifically from the perspective of analyzing the censorship technology used by the censors. Measurement research on censorship is a prerequisite for this approach.

*7.3. Reversible Network Covert Channel.* Many of the CSCs and CTCs proposed alter some traffic features permanently when embedding data. So, they are easy to be detected by ML or DL. Therefore, we can use RDHT (reversible data hiding techniques) to construct the channel which is able to revert the covert traffic to its original form.

*7.4. New Network Environments.* Under some new network environments such as IOT (the Internet of Things), SDN (software defined network), and ICS (industrial control systems), there is no mature covert communication system. With the continuous innovation of new network technologies, we can use the characteristics of the new network to construct more concealed channels.

## 8. Conclusion

As a key technology in the field of network security, covert channels have always been an effective way to protect user privacy. With the development of information steganography and Internet, the network covert channel construction technology continues to be innovated. But at the same time, because the network covert channel can copy with repressive governments, censors are also starting to pay close attention to covert channels. The elimination attack, limitation attack, and detection attack against covert channels have a significant impact on security of covert channels. Although not all attacks can be successful because each channel has different characteristics, there are always one or more attacks that can effectively affect certain performance of covert channel. So, many network covert channels cannot simultaneously satisfy high availability, strong robustness, and high transmission efficiency. The characteristics of the new network can be used to increase the three aspects to some extent.

In this article, we have presented a comprehensive literature review, focusing on the techniques for constructing network covert channels, covert channel metrics, and attacks against network covert channels. The covert channels in the new network environment (streaming media, blockchain, and IPv6) have been introduced and compared. We identify challenges to explore the future direction of improvement and propose possible research methods. We believe this review will contribute to the development of this research area.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. U1636217) and the National Key Research and Development Program of China (Nos. 2016QY05X1000 and 2018YFB1800200), and Key Research and Development Program for Guangdong Province under grant no. 2019B010137003.

## References

[1] X. Xu, Z. M. Mao, and J. A. Halderman, "Internet censorship in China: where does the filtering occur?" in *Proceedings of the 2011 International Conference on Passive and Active Network Measurement*, pp. 133–142, Atlanta, GA, USA, March 2011.

[2] R. Deibert, R. Rohozinski, and A. Manchanda, "Tracking ghostnet: investigating a cyber espionage network," in *Munk Centre for International Studies*, University of Toronto, Toronto, Canada, 2009.

[3] S. Adair, R. Deibert, and R. Rohozinski, "Shadows in the cloud: investigating cyber espionage 2.0, in a joint report of the information warfare monitor and shadowserver foundation," 2010, http://shadows-in-the-cloud.net.

[4] Identify Chinese Cyber Espionage Group, https://tinyurl.com/pntdm64.

[5] S. Aryan, H. Aryan, and J. A. Halderman, "Internet censorship in Iran: a first look," in *Proceedings of the 3rd {USENIX} Workshop on Free and Open Communications on the Internet*, Washington, DC, USA, 2013.

[6] R. Kang, L. Dabbish, and K. Sutton, "Strangers on your phone: why people use anonymous communication applications," in *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, pp. 359–370, San Francisco, CA, USA, 2016.

[7] Y.-F. Li, L.-P. Ding, J.-Z. Wu et al., "Survey on key issues in networks covert channel," *Journal of Software*, vol. 30, no. 8, pp. 2470–2490, 2019.

[8] A. Kwon, H. Corrigan-Gibbs, S. Devadas, and B. Ford, "Atom: scalable anonymity resistant to traffic analysis," 2016, https://arxiv.org/pdf/1612.07841.pdf.

[9] F. Al-Obaidy, S. Momtahen, M. F. Hossain, and F. Mohammadi, "Encrypted traffic classification based ml for identifying different social media applications," in *Proceedings of the 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)*, pp. 1–5, Edmonton, Canada, 2019.

[10] M. Brenner, N. Christin, B. Johnson, and K. Rohloff, Eds., *Lecture Notes in Computer Science*, Springer, Vol. 8976, Berlin, Germany, 2015.

[11] A. M. Piotrowska, J. Hayes, T. Elahi, S. Meiser, and G. Danezis, "The loopix anonymity system," in *Proceedings of the 26th USENIX Security Symposium*, pp. 1199–1216, Vancouver, Canada, August 2017.

[12] S. Rezaei and X. Liu, "Deep learning for encrypted traffic classification: an overview," *IEEE Communications Magazine*, vol. 57, no. 5, pp. 76–81, 2019.

[13] C. Liu, L. He, G. Xiong, Z. Cao, and Z. Li, "FS-NET: a flow sequence network for encrypted traffic classification," in *Proceedings of the 2019 IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pp. 1171–1179, Paris, France, April 2019.

[14] A. C. Aldaya and B. B. Brumley, "When one vulnerable primitive turns viral: novel single-trace attacks on ECDSA and RSA," *IACR Cryptology ePrint Archive*, vol. 2020, p. 55, 2020.

[15] E. S. Alashwali and K. Rasmussen, "What's in a downgrade? A taxonomy of downgrade attacks in the TLS protocol and application protocols using TLS," *IACR Cryptology ePrint Archive*, vol. 2019, p. 1083, 2019.

[16] Eff's Guide, https://www.eff.org.

[17] C. Wang, X. Wang, Y. Lu et al., "Categorization of classic and new covert channel techniques and its application in threat restriction," *Journal of Software*, vol. 31, no. 1, pp. 228–245, 2020.

[18] Q. Tan, J. Shi, B. Fang et al., "Measurement method of unobservability in anonymous communication system," *Computer Research and Development*, vol. 52, no. 10, pp. 2373–2381, 2015.

[19] B. B. Gupta and R. Kumar, "Stepping stone detection techniques: classification and state-of-art," in *Proceedings of the ICRCWP 2015*, Jaipur, India, 2015.

[20] E. Wustrow, C. M. Swanson, and J. A. Halderman, "Tapdance: end-to-middle anticensorship without flow blocking," in *Proceedings of the 23rd {USENIX} Security Symposium*, pp. 159–174, San Diego, CA, USA, August 2014.

[21] I. Cooper and J. Dilley, "Known HTTP proxy/caching problems," *RFC*, vol. 3143, pp. 1–32, 2001.

[22] E. Wustrow, S. Wolchok, I. Goldberg, and J. A. Halderman, "Telex: anticensorship in the network infrastructure," in *Proceedings of the 2011 USENIX Security Symposium*, p. 45, San Francisco, CA, USA, 2011.

[23] R. Dingledine, N. Mathewson, and P. F. Syverson, "Tor: the second-generation onion router," in *Proceedings of the 13th USENIX Security Symposium*, pp. 303–320, San Diego, CA, USA, August 2004.

[24] B. Zantout and R. Haraty, "I2p data communication system," in *Proceedings of 2011 ICN*, pp. 401–409, Toronto, Canada, 2011.

[25] S. Li, M. Schliep, and N. Hopper, "Facet: streaming over videoconferencing for censorship circumvention," in *Proceedings of the 13th Workshop on Privacy in the Electronic Society*, pp. 163–172, Scottsdale, AZ, USA, 2014.

[26] R. McPherson, A. Houmansadr, and V. Shmatikov, "Covertcast: using live streaming to evade internet censorship," *Proceedings on Privacy Enhancing Technologies*, vol. 2016, no. 3, pp. 212–225, 2016.

[27] D. Barradas, N. Santos, and L. Rodrigues, "Deltashaper: enabling unobservable censorship-resistant TCP tunneling over videoconferencing streams," *Proceedings on Privacy Enhancing Technologies*, vol. 2017, no. 4, pp. 5–22, 2017.

[28] Y. Li, L. Ding, J. Wu et al., "Research on a new network covert channel model in blockchain environment," *Journal on Communications*, vol. 40, no. 5, pp. 67–79, 2019.

[29] Botchain Homepage, https://botchain.network/.

[30] D. Frkat, R. Annessi, and T. Zseby, "Chainchannels: private botnet communication over public blockchains," in *Proceedings of the 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), iThings/GreenCom/CPSCom/SmartData 2018*, pp. 1244–1252, Halifax, Canada, July 2018.

[31] N. B. Lucena, G. Lewandowski, and S. J. Chapin, "Covert channels in ipv6," in *Proceedings of the International Workshop on Privacy Enhancing Technologies*, pp. 147–166, Cambridge, UK, 2005.

[32] J. Geddes, M. Schuchard, and N. Hopper, "Cover your ACKs: pitfalls of covert channel censorship circumvention," in *Proceedings of the 2013 ACM SIGSAC Conference on Computer and Communications Security, CCS'13*, Berlin, Germany, November 2013.

[33] A. Dunna, C. O'Brien, and P. Gill, "Analyzing China's blocking of unpublished tor bridges," in *Proceedings of the 8th USENIX Workshop on Free and Open Communications on the Internet (FOCI 18)*, Baltimore, MD, USA, 2018.

[34] N. P. Hoang, S. Doreen, and M. Polychronakis, "Measuring i2p censorship at a global scale," in *Proceedings of the 9th USENIX Workshop on Free and Open Communications on the Internet (FOCI 19)*, Santa Clara, CA, USA, 2019.

[35] T. E. I. Chief and N. D. Fonseca, "Covert channels and countermeasures in computer network protocols," *IEEE Communications Surveys & Tutorials*, vol. 9, no. 3, pp. 44–57, 2007.

[36] B. Lampson, "A note on the confinement problem," 1973, https://www.cs.utexas.edu/~shmat/courses/cs380s_fall09/lampson73.pdf.

[37] G. J. Simmons, "The prisoners' problem and the subliminal channel," in *Advances in Cryptology*, pp. 51–67, Springer, Berlin, Germany, 1984.

[38] J. K. Millen, "20 years of covert channel modeling and analysis," in *Proceedings of the 1999 IEEE Symposium on Security and Privacy*, Oakland, CA, USA, May 1999.

[39] S. Wendzel, S. Zander, B. Fechner, and C. Herdin, "Pattern-based survey and categorization of network covert channel techniques," *ACM Computing Surveys*, vol. 47, no. 3, p. 50, 2015.

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

[41] P. L. Shrestha, M. Hempel, F. Rezaei, and H. Sharif, "A support vector machine-based framework for detection of covert timing channels," *IEEE Transactions on Dependable and Secure Computing*, vol. 13, no. 2, pp. 274–283, 2015.

[42] S. Cabuk, C. E. Brodley, and C. Shields, "Ip covert timing channels: design and detection," in *Proceedings of the 11th ACM Conference on Computer and Communications Security*, pp. 178–187, Washington, DC, USA, 2004.

[43] G. Shah, A. Molina, M. Blaze et al., "Keyboards and covert channels," in *Proceedings of the 2006 USENIX Security Symposium*, vol. 15, Boston, MA, USA, 2006.

[44] C. S. Brodley and E. H. Spafford, *Network covert channels: design, analysis, detection, and elimination*, Ph.D. dissertation, Purdue University, West Lafayette, IN, USA, 2006.

[45] S. H. Sellke, C.-C. Wang, S. Bagchi, and N. Shroff, "TCP/IP timing channels: theory to implementation," in *Proceedings of the IEEE INFOCOM 2009*, pp. 2204–2212, Rio de Janeiro, Brazil, 2009.

[46] X. Li, Y. Zhang, F. Chong, and B. Zhao, "A covert channel analysis of a real switch," Technical report, Department of Computer Science, University of California, Oakland, CA, USA, 2011.

[47] R. Tahir, M. T. Khan, X. Gong et al., "Sneak-peek: high speed covert channels in data center networks," in *Proceedings of the IEEE INFOCOM 2016—The 35th Annual IEEE International Conference on Computer Communications*, San Francisco, CA, USA, 2016.

[48] W. Mazurczyk, M. Smolarczyk, and K. Szczypiorski, "Retransmission steganography and its detection," *Soft Computing*, vol. 15, no. 3, pp. 505–515, 2011.

[49] X. Luo, E. W. W. Chan, and R. K. C. Chang, "TCP covert timing channels: design and detection," in *Proceedings of the 38th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2008*, Anchorage, AK, USA, June 2008.

[50] C. H. Rowland, "Covert channels in the TCP/IP protocol suite," *First Monday*, vol. 2, no. 5, 1997.

[51] Z. Liu, H. Chen, and S. Sun, "Research on covert communication security based on screen content coding," *IEEE Access*, vol. 8, pp. 22275–22280, 2020.

[52] S. Cabuk, C. E. Brodley, and C. Shields, "IP covert timing channels: design and detection," in *Proceedings of the 11th ACM Conference on Computer and Communications Security, CCS 2004*, Washington, DC, USA, October 2004.

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

[54] X. Zhang, L. Zhu, X. Wang, C. Zhang, H. Zhu, and Y.-a. Tan, "A packet-reordering covert channel over volte voice and video traffics," *Journal of Network and Computer Applications*, vol. 126, pp. 29–38, 2019.

[55] K. Ahsan and D. Kundur, "Practical data hiding in TCP/IP," in *Proceedings of the 2002 ACM Workshop on Multimedia Security*, Juan-les-Pins, France, 2002.

[56] X. Zhang, Y.-A. Tan, C. Liang, Y. Li, and J. Li, "A covert channel over volte via adjusting silence periods," *IEEE Access*, vol. 6, pp. 9292–9302, 2018.

[57] C. Krtzer, J. Dittmann, A. Lang, and T. Khne, "WLAN steganography: a first practical review," in *Proceedings of the ACM Multimedia and Security Workshop 2006*, Geneva, Switzerland, 2006.

[58] S. Schulz, V. Varadharajan, and A.-R. Sadeghi, "The silence of the LANS: efficient leakage resilience for IPSEC VPNS," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 2, pp. 221–232, 2014.

[59] R. Rios, J. A. Onieva, and J. López, "Covert communications through network configuration messages," *Computers & Security*, vol. 39, pp. 34–46, 2013.

[60] L. Zhang, T. Huang, W. Rasheed, X. Hu, and C. Zhao, "An enlarging-the-capacity packet sorting covert channel," *IEEE Access*, vol. 7, pp. 145634–145640, 2019.

[61] Z. Trabelsi, W. El-Hajj, and S. Hamdy, "Implementation of an icmp-based covert channel for file and message transfer," in *Proceedings of the 15th IEEE International Conference on Electronics, Circuits and Systems, ICECS 2008*, St. Julien's, Malta, August 2008.

[62] I. J. Kadhim, P. Premaratne, and P. J. Vial, "High capacity adaptive image steganography with cover region selection using dual-tree complex wavelet transform," *Cognitive Systems Research*, vol. 60, pp. 20–32, 2020.

[63] J. Karlin, D. Ellard, A. W. Jackson et al., "Decoy routing: toward unblockable internet communication," in *Proceedings of the 2011 FOCI*, San Francisco, CA, USA, 2011.

[64] A. Houmansadr, T. J. Riedl, N. Borisov, and A. C. Singer, "I want my voice to be heard: IP over voice-over-IP for unobservable censorship circumvention," in *Proceedings of the 2013 NDSS*, San Diego, CA, USA, 2013.

[65] D. Chaum, "Untraceable electronic mail, return addresses and digital pseudonyms," in *Secure Electronic Voting*, pp. 211–219, Springer, Berlin, Germany, 2003.

[66] M. K. Reiter and A. D. Rubin, "Crowds: anonymity for web transactions," *ACM Transactions on Information and System Security*, vol. 1, no. 1, pp. 66–92, 1997.

[67] R. Sherwood, B. Bhattacharjee, and A. Srinivasan, "P5: a protocol for scalable anonymous communication," *Journal of Computer Security*, vol. 13, no. 6, pp. 839–876, 2005.

[68] A. Mani, T. Wilson-Brown, R. Jansen, A. Johnson, and M. Sherr, "Understanding tor usage with privacy-preserving measurement," in *Proceedings of the Internet Measurement Conference 2018*, pp. 175–187, Boston, MA, USA, 2018.

[69] D. Goldschlag, M. Reed, and P. Syverson, "Onion routing," *Communications of the ACM*, vol. 42, no. 2, pp. 39–41, 1999.

[70] A. Kwon, D. Lazar, S. Devadas, and B. Ford, "Riffle: an efficient communication system with strong anonymity," *Proceedings on Privacy Enhancing Technologies*, vol. 2016, no. 2, pp. 115–134, 2016.

[71] D. Chaum, "The dining cryptographers problem: unconditional sender and recipient untraceability," *Journal of Cryptology*, vol. 1, no. 1, pp. 65–75, 1988.

[72] J. McLachlan, A. Tran, N. Hopper, and Y. Kim, "Scalable onion routing with torsk," in *Proceedings of the 16th ACM Conference on Computer and Communications Security*, pp. 590–599, Chicago, IL, USA, 2009.

[73] K. Bauer, D. McCoy, D. Grunwald, and D. Sicker, "Bitblender: light-weight anonymity for bittorrent," in *Proceedings of the Workshop on Applications of Private and Anonymous Communications*, pp. 1–8, Istanbul, Turkey, 2008.

[74] L. Dixon, T. Ristenpart, and T. Shrimpton, "Network traffic obfuscation and automated internet censorship," *IEEE Security & Privacy*, vol. 14, no. 6, pp. 43–53, 2016.

[75] Tor Project Stem, https://stem.torproject.org/.

[76] Tor Project Obfsproxy3, https://gitweb.torproject.org/pluggabletransports/obfsproxy.git/tree/doc/obfs3/obfs3-protocol-spec.txt.

[77] Tor Project Obfsproxy4, https://github.com/Yawning/obfs4/blob/master/doc/obfs4-spec.txt.

[78] B. Wiley, "Dust: A blocking-resistant internet transport protocol," Technical report, University of Texas at Austin, Austin, TX, USA, 2011.

[79] P. Winter, T. Pulls, and J. Fuss, "Scramblesuit: a polymorph network protocol to circumvent censorship," 2013, https://arxiv.org/abs/1305.3199.

[80] H. Mohajeri Moghaddam, B. Li, M. Derakhshani, and I. Goldberg, "Skypemorph: protocol obfuscation for tor bridges," in *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, pp. 97–108, Raleigh, NC, USA, 2012.

[81] Z. Weinberg, J. Wang, V. Yegneswaran et al., "Stegotorus: a camouflage proxy for the tor anonymity system," in *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, pp. 109–120, Raleigh, NC, USA, 2012.

[82] Tor Project Meek, https://trac.torproject.org/projects/tor/wiki/doc/meek.

[83] C. Gülcü and G. Tsudik, "Mixing e-mail with babel," in *Proceedings of the 1996 Symposium on Network and Distributed Systems Security*, San Diego, CA, USA, February 1996.

[84] G. Danezis, R. Dingledine, and N. Mathewson, "Mixminion: design of a type III anonymous remailer protocol," in *Proceedings of the 2003 IEEE Symposium on Security and Privacy*, Berkeley, CA, USA, May 2003.

[85] L. Wang, K. P. Dyer, A. Akella, T. Ristenpart, and T. Shrimpton, "Seeing through network-protocol obfuscation," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, Denver, CO, USA, 2015.

[86] S. T. Ali, P. McCorry, P. H.-J. Lee, and F. Hao, "Zombiecoin 2.0: managing next-generation botnets using bitcoin," *International Journal of Information Security*, vol. 17, no. 4, pp. 411–422, 2018.

[87] Yang, "Research on network concealed channels in ipv6," *Journal of Southeast University*, vol. 37, no. s1, pp. 141–148, 2007.

[88] J. Ullrich, K. Krombholz, H. Hobel, A. Dabrowski, and E. Weippl, Ipv6 security: attacks and countermeasures in a nutshell," in *Proceedings of the 8th USENIX Workshop on Offensive Technologies (WOOT 14)*, San Diego, CA, USA, 2014.

[89] ISATAP, https://zh.wikipedia.org/zh-hans/ISATAP.

[90] 6to4, https://zh.wikipedia.org/wiki/6to4.

[91] 6over4, https://zh.wikipedia.org/wiki/6over4.

[92] M. K. Reiter and A. D. Rubin, "Anonymous web transactions with crowds," *Communications of the ACM*, vol. 42, no. 2, pp. 32–48, 1999.

[93] O. Berthold, A. Pfitzmann, and R. Standtke, "The disadvantages of free mix routes and how to overcome them," *Designing Privacy Enhancing Technologies*, vol. 63, no. 164, pp. 30–45, 2001.

[94] C. Diaz, S. Seys, J. Claessens, and B. Preneel, "Towards measuring anonymity," in *Proceedings of PET 2002*, San Francisco, CA, USA, 2002.

[95] H. W. Lilliefors, "On the Kolmogorov-smirnov test for normality with mean and variance unknown," *Journal of the American Statistical Association*, vol. 62, no. 318, pp. 399–402, 1967.

[96] R. Archibald and D. Ghosal, "A covert timing channel based on fountain codes," in *Proceedings of the 11th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom 2012*, Liverpool, UK, June 2012.

[97] S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[98] D. Barradas, N. Santos, and L. E. T. Rodrigues, "Effective detection of multimedia protocol tunneling using machine learning," in *Proceedings of the 27th USENIX Security Symposium 2018*, pp. 169–185, Baltimore, MD, USA, August 2018.

[99] R. Archibald and D. Ghosal, "A comparative analysis of detection metrics for covert timing channels," *Computers & Security*, vol. 45, pp. 284–292, 2014.

[100] J. Wu, Y. Wang, L. Ding, and X. Liao, "Improving performance of network covert timing channel through Huffman coding," *Mathematical and Computer Modelling*, vol. 55, no. 1-2, pp. 69–79, 2012.

[101] J. Wu, L. Ding, and Y. Wang, "Research on key problems of covert channel in cloud computing," *Journal of China Institute of Communications*, vol. 32, no. 9, 2011.

[102] S. Gianvecchio and H. Wang, "Detecting covert timing channels: an entropy-based approach," in *Proceedings of the 14th ACM Conference on Computer and Communications Security*, pp. 307–316, Alexandria, VA, USA, 2007.

[103] A. Porta, G. Baselli, D. Liberati et al., "Measuring regularity by means of a corrected conditional entropy in sympathetic outflow," *Biological Cybernetics*, vol. 78, no. 1, pp. 71–78, 1998.

[104] A. Houmansadr and N. Borisov, "Coco: coding-based covert timing channels for network flows," in *Proceedings of the 13th International Conference on Information Hiding*, Prague, Czech Republic, May 2011.

[105] P. Wang, S. Lan, J. Zhang, and G. Liu, "Covert timing channel method based on TCP timestamp option," *Journal of PLA University of Science and Technology*, vol. 16, no. 2, pp. 120–125, 2015.

[106] G. He, M. Yang, J. Luo, and L. Zhang, "Online identification of TOR anonymous communication traffic," *Journal of Southeast University*, vol. 24, no. 3, pp. 540–556, 2013.