Systematic Analysis on the Effectiveness of Covert Channel Data Transmission

Abdulrahman Alhelal, Mohammed Al-Khatib
Computer Science Department, Imam Mohammad Ibn Saud Islamic University, Riyadh, Saudi Arabia

***Abstract*—A covert channel is a communication channel that allows parties to communicate and transfer data indirectly. Covert channel types are storage, timing, and behavior channels. Covert channels can be used for malicious and benign applications. A covert channel is a mechanism for violating the communication security policy that was not anticipated by the system creator. Recently, covert channels are used to transfer text, video, and audio information between entities. This article, discusses studies related to the development of covert channels as well as the research works that focus on improving the performance/throughput of covert channels. Also, it analyzes the previous studies in terms of publication type, year of publication, article title, article purpose, transferring file format used in covert channel, coding technique, throughput performance, time needed to transfer files, and article limitations.**

Keywords*—Covert channel; transmission; limitations; file; encoding throughput; performance; time; audio; video; text

I. Introduction

A covert channel is a mechanism for violating the communication security policy, where it is employed for encoding and decoding [1, 2]. The most important channel types are storage, time, and behavior channels. Storage channels is created using disk, physical memory to be shared between entities during using covert channel. A timing channel is a communication channel that can send/receive data by altering an entity's timing behavior. Covert channel works by altering the behavior of an application. A covert channel is used in data transmission in order to send different types of data between entities. There are several restrictions in the transmission process, such as channel capacity in terms of throughput. This article concentrates on investigating and evaluating the covert channels utilized in data transmission. Then analyze the collected studies about covert channels to identify the mechanisms used to create covert channels. In addition to determine, the throughput needed to transfer files. A comparative survey of related works is done in terms of several dimensions: The technique used to create the channel, the purpose of the channel, data format, coding techniques, throughput performance, time needed to transfer files, and article limitations. The methodology for conducting a comparative analysis about covert channel studies, which concern in data transmission, is shown in Fig. 1.

The rest of the article is structured as follows: Section II provides background and basic knowledge about covert channels, covert channel and scenarios. Section III discusses and analyzes related work about the effectiveness of covert channel data transmission. Section IV discusses the comparative analysis for related works studies and article recommendations. Section V concludes the article and lists future works.

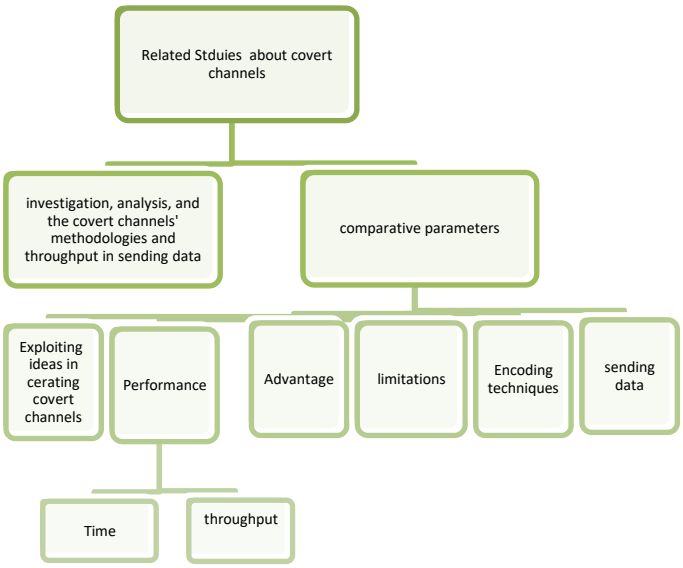

Fig. 1. Comparative survey methodology.

II. Background of Covert Channel

This section will cover the covert channel, its types, its scenarios, and covert channel applications.

A. Covert Channel

Lampson [3], first one that identifies a covert channel in a single machine computer environment in 1973 in his research titled "A Comment on the Confinement Problem". He describes a covert channel as a channel that is not intended for the transmission of data. Covert channel is a communication channel that allows parties to communicate and transfer data indirectly using per-agreement knowledge. As depicts in Fig. 2, the covert cannel model uses to encode and decode the original messages. Covert channel is a mechanism for violating the communication security policy that was not anticipated by the system creator [1, 2, 3].

The sender and receiver (e.g.; Alice and Bob respectively) want to communicate covert massage in spite of attacker existence (e.g.; Wendy). A covert channel is a type of secret channel. It sends secret data in an unseen manner by the monitoring system. Covert channel compromises the normal communication connection. Assume that Alice and Bob are connected through networked computers under the supervision of a network administrator.

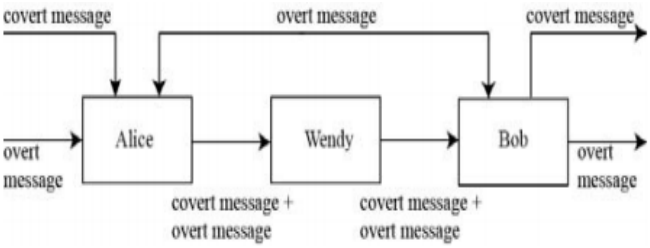

Fig. 2. Covet channel model [2].

As a result, a network covert channel exists when Alice and Bob create a concealed communication channel that is undetectable by the network monitoring system. Alice and Bob must have pre-shared key called pre-agreement knowledge. So the channel is ordinal channel that has covert data from Alice to Bob. Secret data is agreed upon by the entities as pre-agreement knowledge that is used to encode and decode the original messages. For example, if the pre-agreement between client and server is that each word with an even number of letters is read as a 1 and each word with an odd number of letters is read as a 0. For example, if a client sends a message to the server, the "covert channel", the server will interpret it as a "10" [1, 2, 4].

In the communication process, the secure transmission of secret communications relates to two aspects: communication content security and communication connection security. The security of these two features can be improved by using network covert channels [5]. A covert channel can be either a standalone or a networked system. The covert information is exchanged between elements in the stand-alone system. The covert information is sent over the network in a network–based system [6]. Initially, researchers focus on local covert channels, in which two processes with differing levels of security might connect with one another to leak information. Typically, a process with a high security level leaks information to a process with a low security level. With the growth and rapid development of computer networks, the focus has switched to network covert channels, which can embed covert information into network protocols [7].

B. Covert Channel Types

The most important three categories of covert channels are storage covert channels, timing covert channels, and behavior-based covert channels [3, 8, 9, 10]. Fig. 3 shows the main three types of covert channels.

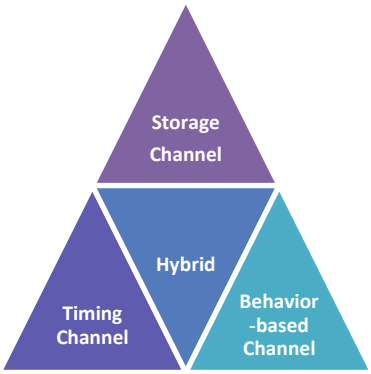

Fig. 3. Types of covert channel.

1) Storage covert channels: The sender and receiver create covert channel based on shared storage data agree on shared data. To embed covert data, storage channels primarily modify two characteristics of information.

a) File names can be utilized as entity attributes for storage channels. It can be altered by a single process. When any process performs a read operation, a message transfer between the processes occurs. It's also possible to change file attributes, which are properties of a file. Even though requesting a file that does not exist, the file system's feedback status can be used for storage channels [10, 11].

b) Shared resources: Storage channels can be made up of disk blocks, physical memory, I/O buffers, allocated I/O devices, and other queues for shared devices like printers and plotters. Storage channels are defined as a system feature that allows one system entity to signal information to another entity by writing to a storage location that is then read by the second entity directly or indirectly. It encrypts data and stores it on a medium that is shared by the endpoints. The shared resource was not made to transport data. A covert channel can be implemented in a networked system by utilizing reserved fields in various packet headers/footers or by hiding data in the payload. Attackers employ storage channels to encrypt information being transferred and then decode it afterwards. Some fields in TCP/IP stacks are left empty or unused. These vacant fields are used by attackers as storage conduits [10, 11].

2) Timing channel: A timing channel is a communication channel that can send data to a receiver/decoder by altering an entity's timing behavior. For example, packet delays between packet streams, packet reordering inside packet streams, or encoder resource access time. allows one system entity to communicate with another by manipulating its own use of a system resource in a way that affects the second entity's system response time. It changes the event time so that data can be shared between the endpoints. The main benefit is that the covert channel may be set up without affecting the transmission data stream. To this purpose, timing-based channels can be used with a variety of protocols because they are not affected by the network's packet syntax or semantics.
Attackers employ timing covert channels to modify system resources in order to deliver a message over time. The temporal delay between consecutive packet transfers is known as inter-packet delay. Timing-based covert channels are suitable for designing a complementary nonce synchronization channel that can improve robustness against message loss in existing authentication techniques, according to Vanderhallen et al. [11]. They tested this approach on top of an open-source authenticated CAN communication library, demonstrating that covert timing channels can improve communication robustness in benign situations while without compromising the security guarantees of the underlying authentication primitives when attacked.

3) Behavior-based covert channels: Behavior-based covert channels are described as a communication channel in which the sender or receiver's internal states are modulated by

purposefully selecting certain inputs to the systems. It works by altering the behavior of an application on purpose. The endpoints use this change to communicate. Covert channels based on behavior address the application level and are neither synchronized or dependent on a specific network protocol. As a result, they are more difficult to avoid and detect: identifying the message in a covert channel requires complete understanding of the application. Tamer Fatayer et al. [4, 9, 12], proposed behavior based channel through exploiting memory address in Linux operating system, where this channel change the behavior of using system calls and redirect the system call to malicious code implement by attacker to perform specific tasks.

C. Covert Channel Senarios

Zaider et al. [2] mention that that are different applications or usages (legitimate and illegitimate) for cover channel. Many covert channel applications are harmful or undesirable, which poses a severe danger to network security [2, 13, 14, 15, 16]. Malicious covert channel applications compromise network security. The Internet is the ideal high-bandwidth medium for covert communications because of the massive amount of information it contains as well as the enormous variety of data protocols. It is essential to comprehend current covert channel strategies while creating countermeasures. Identification, elimination, and capacity restriction of covert channels must be addressed to protect future computer networks, which is difficult. The current trend of covert channels is used in transforming secret information between entities [4, 9, 12, 17, 18, 19]. The researchers exploit network to transfer secret data (e.g., text, video, web, and audio) between entities. There are various covert channel scenarios:

1) Storage channel scenarios: Fatayer [12] developed a covert channel through developing table that is considered as pre-agreement data between entities. The covert channel-using table to transmit secure data that allows two entities to agree on a secret key using encryption to prevent an attacker from getting any information. Qiumin Xu et al. [20] a Trojan application can cause resource contention by altering the contents of a cache set to encode '1' and leaving the resource idle to encode '0'. On the other hand, the Spy application visits the cache and measures its access time in order to decode the transferred bit. Similarly, a Trojan application can cause contention by consuming a lot of execution units, warp schedulers, and instruction fetch units to encode '1' and then leaving those resources idle to encode '0,' which the spy can decode.

*2) Timing channel scenario:*Timing channels seek for TCP segments, which provide a finer range of information encoding options. For instance, steganography methods may take use of ACK/SYN sequences [21]. Information-containing segment patterns and artificial reordering. Following the same patterns as legal traffic and are immune to regularity testing. Active timing channels, on the other hand, create traffic and may easily maintain the form of the distribution, making them less susceptible to shape detection tests. However, they are unable to maintain pattern recurrence and are easily detected using a regularity test.

3) Behavior-based scenario: Fatayer et al. [9] exploit buffer overflow vulnerability in C language on the Linux operating system to develop a covert channel. They exploit stack-overflow attacks vulnerability and address space layout randomization on Linux to transmit different file formats between entities. On entity tries to guess the randomization value that cause buffer overflow in Linux memory, while the other entity monitoring the count the guessing numbers till the success guess.

III. Literature Review Analysis

This article will discuss studies that target covert channels utilized in data transmission between entities in this section. After that, we will analyze those studies to investigate the shortcomings and limitations of cover channel throughput during sending data by addressing these restrictions and limitations.

A. Covert Channels in Data Transmission

Channel in a wireless communication system with adaptive rate: They were able to effectively demonstrate a covert channel with a throughput of more than 150 Mbps that reliably delivered the hidden payload while minimizing the mistakes seen in the underlying communications system. Although the focus of this study was on IEEE 802.11ad, additional adaptive rate communication protocols should be able to benefit from using modulation and coding schemes selection to increase covert channel capacity. The selection of a cover object is limited to objects that can tolerate a certain amount of distortion (e.g., Audio and video), which is a significant flaw in this method. They don't used text or executable files. To minimize distortion on the underlying communications channel without compromising the covert channel's throughput, a modified embedding mechanism was developed.

Wendzel et al. [22] introduce a full survey covert channel that used to hide information inside network protocol. They investigated and analyzed around 109 techniques targeting covert channels that hide communication protocol. They classified a covert channel according to special pattern. They classify the covert channel according to eleven patterns which are Size Modulation, sequence of header/PDU elements to encode hidden information, add redundancy, PDU corruption/Loss pattern, random value pattern, value modulation pattern, reserved/unused pattern (a reserved or unused header/PDU element was used by the covert channel to encode data), inter-arrival time pattern, data rate of a traffic, PDU order patter, and retransmission pattern as depicts in Fig. 4.

In this survey we concerned with data rate of a traffic pattern, which allows covert channel sender alters the data rate of a traffic flow to the covert channel receiver. Using exception handling, we will construct a covert channel that we will utilize to deliver files to the server. Additionally, we employed a pattern called program flow pattern, which modifies program execution and transmits covert data to the receiver.

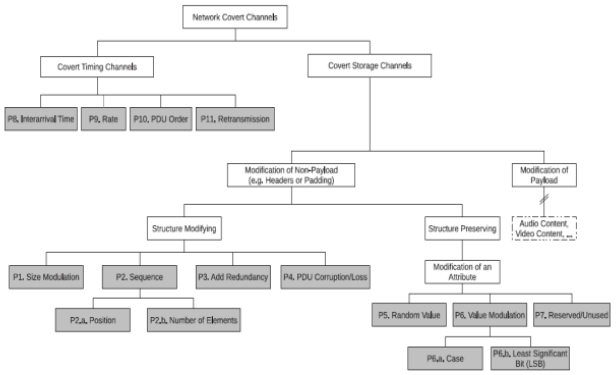

Fig. 4. Classification pattern of covert channel [22].

Zander et al. [22] presented a survey of developing network covert channels. The terms, adversary scenario, covert channel techniques, and countermeasures are all covered in this survey. Covert channels are used in sending information through network protocols. The survey showed the covert channel may use unused header bits, header extensions, padding, the IP Identifier and Fragment Offset, the TCP Initial Sequence Number (ISN), checksum fields, the time to live (TTL) field, the modulation of address fields and packet lengths, the modulation of timestamp fields, packet rate and timing, message sequence timing, packet loss and packet sorting, frame collisions, and ad hoc routing protocols. Also, they mention the counter measures techniques that used to detect and prevent covert channel. The following countermeasures are used such as:

1) Eliminate the channel including host security, Network security, traffic normalization.

2) Limit the bandwidth of the channel.

They introduced [22] an explanation of covert channel capacity, the amount of information that can be encoded in a resource's size (storage channels) or the speed at which it can be modulated (timing channels) can be used to estimate the covert channel capacity. Estimating the capacity in terms of bits per packet or bits per message sequence is simple for some channels. The amount of overt communication between covert sender and receiver or the amount of acceptable overt traffic accessible in the network determines capacity in bits per second. Therefore, it must modify the mechanisms for generating and producing covert channels in order to boost their capacity.

Schmidbauer et al. [18] present two covert channels that exploit nonce-based network authentication. First covert channel exploit key-based authentication and the second covert channel exploit hash-based authentication. These channels are used for sending encrypted information between parties. They investigated and exploited the challenge-response authentication with a nonce for transfer secret information. They evaluated their covert channel in terms of throughput rate. They increased the performance of the throughput by applying Compression and codebook techniques. They measure the throughput through number of attempt to achieve challenge-response authentication mechanism. Fig. 5 shows that using a compressed JS-Hex file requires 500 attempts to communicate 4 bytes over Hash-based covert channel, whereas using an uncompressed JS-Hex file requires 3000 attempts.

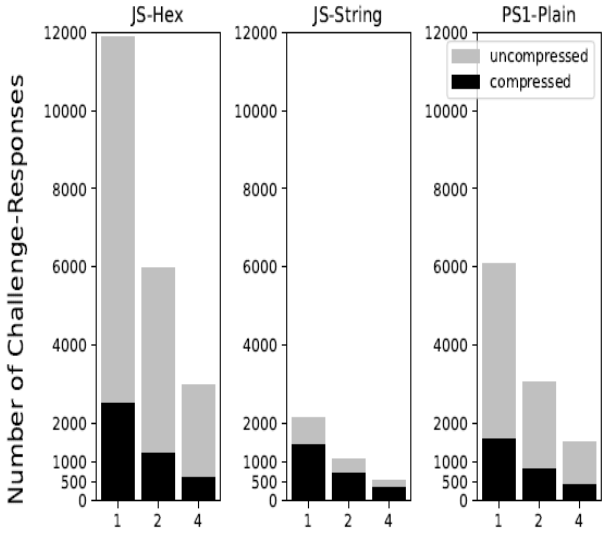

Fig. 5. Throughput hash-based covert channel [18].

Fatayer [12] developed a storage covert channel through developing a storage memory to be a pre-agreement data store between entities. Data structures are utilized to establish the covert channel, which will be used to transfer secure data between entities. This covert channel enabled parties to agree on a secret key. This mechanism is flexible in generating key with different size. On the other hand, it cost more traffic overheads and time consuming.

Fatayer et al. [9] exploit stack-overflow attacks and address space layout randomization on Linux to transmit different file formats. The sender tries to guess the delta_mmap memory (length 16 bit). One entity wants to send Files indirectly to other entity through a server. They are located in the same machine. Client tries to guess the random memory offset of standard C-library (delta_mmap) of the vulnerable server, and Bob tracks Alice's guessing attempts by monitoring the server process. Information to be sent is encoded in the number of failed guesses before success. They transmitted text and audio files between entities. They achieve best throughput performance using hexadecimal coding as depicts in Table I.

TABLE I. COVERT CHANNEL PERFORMANCE DURING TRANSFERRING AUDIO FILES [9]

Audio file size	*Optimization time*	*Without optimization*
2.3 KB	.76 minute	1.01 minute
1.2 MB	91. 9 minute	234 minute

They suggested researchers to use encoding techniques to enhance the throughput performance. Additionally, they will look into methods for boosting channel throughput by determining the ideal block size, ideal number of tables, and ideal region number encoding—possibly utilizing Huffman-code. This article contribution is stemmed from this point. This article will develop covert channel to send file with different format with coding mechanism such as Byte, Hexadecimal, Huffman, and base64 coding.

Elsadig et al. [17], introduce a full comprehensive survey about the usage of covert channels and their types. They mention several benign usages of the covert channel, where the author considers it a new direction in security. Previous works focus on developing covert channels that depend on a new idea and the advantages and disadvantages of existing covert channels. There researches mention that there is limitation in throughput performance. Few research works investigated the use of covert channels to send files with different formats. Moreover, the performance of such covert channels has not been adequately explored. Previous researches did not investigate the use of encoding algorithms to improve the throughput performance of covert channels as well.

Jens et al. [19] exploit VoIP communications as a technique to increase privacy in sending files. They suggested hiding traffic within VoIP conversations to prevent disclosure from blocking the ongoing exchange of information. They use the voice activity detection features which found in client interfaces to create phony quiet packets that may be utilized as a carrier for transferring secret data. Results show that the suggested method may be effective for enforcing privacy in practical use scenarios, particularly for file transfers. They leveraged VoIP traffic by developing a virtual network interface for tunneling protocols of the TCP/IP suite.

Privacy Enhancing Technology Voice Activity Detection (PETVAD) is slower than a direct access because HTTP is influenced by the TCP's subpar performance when there are significant delays. Part of results in [19] as depicted in Table II, which indicates that to send 1 MB webpage it cost 283 (KB\s) as throughput and 3.61 second in the existence of 100ms delays [19]. Throughput is measure through constructing local area and wide area network configurations. They used three webpages to be sent through the covert channel. Each page has 1, 10, and 100 inline objects, each of which is 1MB, 100 KB, and 10 KB in size.

Cumulative Distribution Functions (CDFs) that describe the probability distribution of random variables of transfer over times. As it is visible, the larger the size of the website, the higher the variations experienced by the users when retrieving a page as shown in Fig. 6. The low bandwidth may cause interactive services (like web surfing) to lag too much, hence some sort of optimization or content scaling may be advised in these situations.

TABLE II. TIME AND THROUGHPUT FOR DIRECT AND PETVAD COVERT CHANNELS [19]

web pages size	*Time (second)*	*Rate (KB/s)*	*Techniques*	*Delay*
1* 1MB	3.61	283	Direct	100 ms
10* 100K	4.98	205		
100 * 10 K	21.65	47		
1* 1MB	422.21	2.43	PETVAD	
10* 100K	442.53	2.31		
100 * 10 K	660.53	1.55		

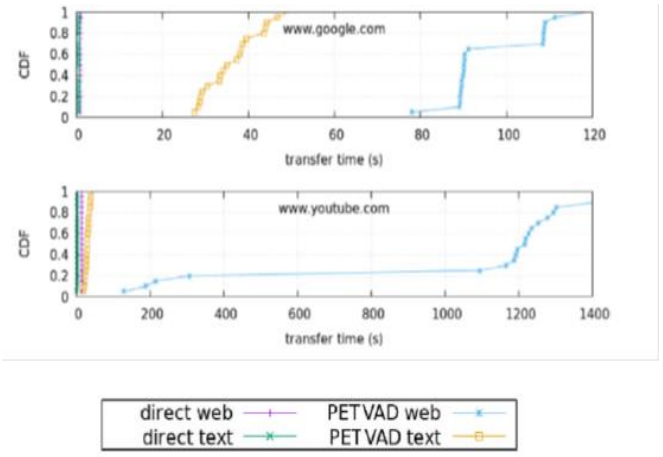

Fig. 6. Using a VIOP covert channel to transfer web sites [19].

B. Analysis Summary

The covert channels utilized for data transfer between entities are the topic of this article. There are several methods for creating covert channels, each of which can be used to encode and decode information. There are several types of information that are sent through channels, such as text files, audio, video, and web pages. We analyze and evaluate the performance of the covert channel in terms of throughput. The term "throughput" in this article means the number of transmitted bits per second. We observed from related works that the covert channel has low throughput. The authors use several mechanisms to enhance the performance, such as using encoding techniques.

IV. DISCUSSION ANALYSIS AND RECOMMENDATION

A. Systetic Review Discussion

This article discusses and analyzes studies related to the development of covert channels as well as the research works that focus on improving the performance/throughput of covert channels. This article analyzes the previous studies in terms of publication type, year of publication, article title, article purpose, transferring file format used in covert channel, coding technique, throughput performance, time needed to transfer files, and article limitations. Then compare conventional and modern mechanisms for creating covert channels. Table III, shows that covert channel is used for transferring files between entities. There are several mechanisms for creating covert channels. Current trends for creating covert channel through exploiting: Networks protocol (e.g., TCP), System interfaces, and Programming language.

Table III, examines previous studies to investigate covert channel capacity, throughput, and data transfer time; a comparative survey of related work in terms of exploiting ideas to create covert channels, channel type (e.g., storage channel), creation purpose (malicious and benign), data transmission type (e.g., audio and video), measurement performance, optimization techniques, advantages, and drawbacks where it's clear that the covert channel can be used to send different type of data such as audio, video, webpages, and text. The covert channel must be supported with coding techniques to increase the channel throughput. The covert channel must authenticate

parties before using it to send data. Furthermore, one of the primary goals of the covert channel is to improve throughput performance.

TABLE III. COVERT CHANNELS OF PREVIOUS STUDIES

Terms	***Collected research***
Exploiting idea	Ideas of collected article [4, 18, 19, 22, 23, 24] exploits networks protocol (e.g., TCP), system interfaces
Channel Type	Storage channel, timing channel, program flow
Purposed	Most of article s target is attack purposes. Some of them used to send secret data.
File Type	Text file, webpages and voice [19],
Measurement performance	Articles measure throughput and time of sending files performance.
Coding Optimization	Hex-coding [12], compression and code book technique [18], without coding [19].
Advantage	Every article is consideration as a new idea for a covert channel. These article s serve as warnings about the threats against network and software.
Drawbacks	The throughput performance is still low and needed to be improved.

Evaluating and analyzing the previous related works is depicted in Table IV in terms of publication type (conference or journal), publication year, article title, main idea of generating a covert channel, type of data transmission, coding techniques used in generating data, throughput performance in sending data, time performance in sending data, and article covert channel limitations.

TABLE IV. (A). THE MAIN COMPARISON POINTS FOR PROPOSED COVERT CHANNEL AND PREVIOUS COLLECTED ARTICLES

article reference	***Publication type/year***	***Article Name***	***Main idea***	***Data transfer type***
[18]	*Conference/* 2022	"Challenging channels: Encrypted covert channels within challenge-response authentication.	They investigated and exploited the challenge-response authentication with a nonce for transfer secret information	They mention JS hexadecimal, JS-string, JS-plain, And ASCII alphabet
[4, 9, 12]	Conference/ 2011	OverCovert: Using Stack Overflow Software Vulnerability to Create a Covert Channel	Exploiting stack memory to hacks Linux memory functions to create covert channel.	Text and audio file
[19]	*Journal/ 2020*	VoIP network covert channels to enhance privacy and information Sharing	Develop covert channel depends on using the voice activity detection features which found in client interfaces to create phony quiet packets that may be utilized as a carrier for transferring secret data	Web pages as Google YouTube
proposed	*2022*	Exploiting a program execution for developing a high throughput covert channel.	exploit program execution	Text, audio, video,

TABLE V. (B). THE MAIN COMPARISON POINTS FOR OUR PROPOSED COVERT CHANNEL AND PREVIOUS COLLECTED ARTICLES

article reference	***Coding***	***Throughput Concept***	***Time***	***Limitation***
[18]	compression and codebook techniques	Number of attempts Needs to achieve challenge-response authentication	Not computed	countermeasures that allow the limitation of bandwidth, for example the utilization of appropriate RSA keys, and the elimination of the CC through the deployment of application-level firewalls
[4, 9, 12]	Used just hexadecimal coding	Number of bit sending during time	Time for agreement time and time for sending different files	Throughput is still low and authors just used hexadecimal . They presumptively believe that a server program is monitored by a local program that resides on the server's computer. They presume that the server cooperate with two parties.
[19]	There no special coding it just using two networks	Number of object(KB) in web pages transmitted per second	Computed	content-rich web browsing may require further optimizations, such as operating in a text-only fashion

B. Recommendation

Through related works analysis and developing covert channel, the following recommendations are

1) To develop covert channel countermeasure, we must understand covert channel mechanisms in terms of identification, elimination, and the capacities.

2) Applications and scenarios for covert channels in computer networks are varied.

3) Increased channel capacity can be achieved by changing the processes used to create and develop covert channels.

4) Both beneficial and harmful purposes are carried out through the covert channel.

5) The researchers concentrate on the fact that the channel capacity has a gap, which has to be filled by future researchers.

6) Using coding techniques maybe increase the throughput of channel in sending data.

7) Sending information through covert channel need security issues (e.g., entities authentication) beside sending data covertly.

V. Conclusion

Covert channel has several types: storage, timing, and behavior channels. Covert channel has several applications including data transmission. In this article, we focused on investigating and assessing the covert channels that used in data transfer. We collected earlier studies to identify methods for creating covert channels and how they measured the throughput at which data was delivered between organizations. We do a comparative survey of related work in terms of exploiting ideas to create covert channels, channel type (e.g., storage channel), creation purpose (malicious and benign), data transmission type (e.g., audio and video), measurement performance, optimization techniques, advantages, and drawbacks. We determine recommendations based on conducted survey, including: Applications and scenarios of covert channels in computer networks are varied. Increased channel capacity can be achieved by changing the technique that used to create and develop covert channel. In covert channel, coding techniques increase throughput of channel in sending information through covert channel need security issues (e.g., entities authentication) besides sending data covertly. The covert channel must be supported with coding techniques to increase the channel throughput. The covert channel must authenticate parties before using it to send data. Furthermore, one of the primary goals of the covert channel is to improve throughput performance.

Acknowledgment

I am a student at Imam Mohammed Ibn Saud Islamic University. I extend my appreciation to the Deanship of Scientific Research at Imam Mohammed Ibn Saud Islamic University for funding and supporting this work through the graduate student research support program. I would like to express my sincere gratitude to my advisor, Prof. Mohammed Al-Khatib, for their invaluable guidance and support throughout my master's program. Their expertise and encouragement helped me to complete this research.

References

[1] Elsadig, M. A., & Fadlalla, Y. A. (2018). Packet length covert channels crashed. *J Comput Sci Comput Math*, *8*(4), 55-62.

[2] Zander, S., Armitage, G., & Branch, P. (2007). A survey of covert channels and countermeasures in computer network protocols. IEEE Communications Surveys & Tutorials, 9(3), 44-57.

[3] Lampson, B. W. (1973). A note on the confinement problem. Communications of the ACM, 16(10), 613-615.

[4] Xing, J., Kang, Q., & Chen, A. (2020, January). Netwarden: Mitigating network covert channels while preserving performance. In USENIX Security.

[5] Tian, J., Xiong, G., Li, Z., & Gou, G. (2020). A survey of key technologies for constructing network covert channel. Security and Communication Networks, 2020, 1-20.

[6] Dakhane, D. M., & Deshmukh, P. R. (2015, January). Active warden for TCP sequence number base covert channel. In 2015 International Conference on Pervasive Computing (ICPC) (pp. 1-5). IEEE.

[7] Wendzel, S., Zander, S., Fechner, B., & Herdin, C. (2015). Pattern-based survey and categorization of network covert channel techniques. ACM Computing Surveys (CSUR), 47(3), 1-26.

[8] Alcaraz, C., Bernieri, G., Pascucci, F., Lopez, J., & Setola, R. (2019). Covert channels-based stealth attacks in industry 4.0. IEEE Systems Journal, 13(4), 3980-3988.

[9] Fatayer, T. S., Khattab, S., & Omara, F. A. (2011, February). OverCovert: Using stack-overflow software vulnerability to create a covert channel. In *2011 4th IFIP International Conference on New Technologies, Mobility and Security* (pp. 1-5). IEEE.

[10] Gasser, Morrie. Building a secure computer system. New York: Van Nostrand Reinhold Company, 1988.

[11] Vanderhallen, S., Van Bulck, J., Piessens, F., & Mühlberg, J. T. (2021). Robust authentication for automotive control networks through covert channels. *Computer Networks*, *193*, 108079.

[12] Fatayer, T. S. (2020). Secure Communication Using Cryptography and Covert Channel. In Computer and Network Security. IntechOpen.

[13] Giffin, J., Greenstadt, R., Litwack, P., & Tibbetts, R. (2003). Covert messaging through TCP timestamps. In Privacy Enhancing Technologies: Second International Workshop, PET 2002 San Francisco, CA, USA, April 14–15, 2002 Revised Article s 2 (pp. 194-208). Springer Berlin Heidelberg.

[14] Dong, P., Qian, H., Lu, Z., & Lan, S. (2012). A Network Covert Channel Based on Packet Classification. *Int. J. Netw. Secur.*, *14*(2), 109-116.

[15] Yuan, B., & Lutz, P. (2005). A covert channel in packet switching data networks.

[16] Akhtari, S., Moghim, N., & Mahdavi, M. (2020). Middleman covert channel establishment based on MORE routing protocol using network coding in ad hoc networks. *International Journal of Communication Systems*, *33*(7), e4320.

[17] Elsadig, M. A., & Fadlalla, Y. A. (2016). Survey on covert storage channel in computer network protocols: detection and mitigation techniques. *International Journal of Advances in Computer Networks and Its Security*, *6*(3), 11-17.

[18] Schmidbauer, T., Keller, J., & Wendzel, S. (2022, August). Challenging channels: Encrypted covert channels within challenge-response authentication. In *Proceedings of the 17th International Conference on Availability, Reliability and Security* (pp. 1-10).

[19] Saenger, J., Mazurczyk, W., Keller, J., & Caviglione, L. (2020). VoIP network covert channels to enhance privacy and information sharing. *Future Generation Computer Systems*, *111*, 96-106.

[20] Xu, Q., Naghibijouybari, H., Wang, S., Abu-Ghazaleh, N., & Annavaram, M. (2019, June). Gpuguard: Mitigating contention based side and covert channel attacks on gpus. In *Proceedings of the ACM International Conference on Supercomputing* (pp. 497-509).

[21] Nowakowski, P., Zórawski, P., Cabaj, K., & Mazurczyk, W. (2020, August). Network covert channels detection using data mining and

hierarchical organisation of frequent sets: an initial study. In *Proceedings of the 15th International Conference on Availability, Reliability and Security* (pp. 1-10).

[22] Zander, S., Armitage, G., & Branch, P. (2007). A survey of covert channels and countermeasures in computer network protocols. *IEEE Communications Surveys & Tutorials*, *9*(3), 44-57.

[23] Cabuk, S., Brodley, C. E., & Shields, C. (2004, October). IP covert timing channels: design and detection. In *Proceedings of the 11th ACM conference on Computer and communications security* (pp. 178-187).

[24] Berk, V., Giani, A., & Cybenko, G. (2005). Detection of covert channel encoding in network packet delays.