Università di Genova | DIPARTIMENTO INFORMATICA, BIOINGEGNERIA, ROBOTICA E INGEGNERIA DEI SISTEMI

Computer Science Workshop
PhD program in Computer Science and Systems Engineering

Are We Protected Against Network Covert Channels?

Marco Zuppelli

Motivation and Context

Information-hiding-based techniques like *covert channels* are increasingly used by attackers to conceal malware in different carriers, such as images or inter-process communication services. These techniques allow, for example, to secretly exfiltrate information, elude well-know detection mechanisms, or remotely activate a backdoor [1]. Among the different carriers, the usage of network traffic features is appealing to attackers, as they offer a wide range of possibilities. For instance, *network covert channels* can be created by directly concealing malware or malicious commands in header fields of some protocol, e.g., Time To Live of IPv4 or Traffic Class of IPv6, or by encoding secret data in the temporal evolution of traffic, e.g., in the inter-packet time.

The adoption of network covert channel often leads to security problems: in fact, several out-of-the-box Intrusion Detection Systems (IDSes), or firewalls do not consider them as a major threat. Moreover, extending the functionalities of network security tools could lead to inefficient behaviours since each protocol requires ad-hoc rules. Being able to spot covert channels is mandatory to fully assess the security capabilities of a network infrastructure.

In this poster, we will answer the question of whether we are really protected from the threat of network covert channels, by assessing the detection capabilities of the most popular and open source security mechanisms, i.e., Snort, Zeek and Suricata.

Attack Model and Challenges

Figure 2 depicts the reference scenario. The Covert Sender (e.g., a compromised host) wants to secretly communicate with the Covert Receiver (e.g., a Command & Control server) to exfiltrate sensitive data or activate a backdoor. To do this, they aim at bypassing the Firewall by implementing a network covert channel.

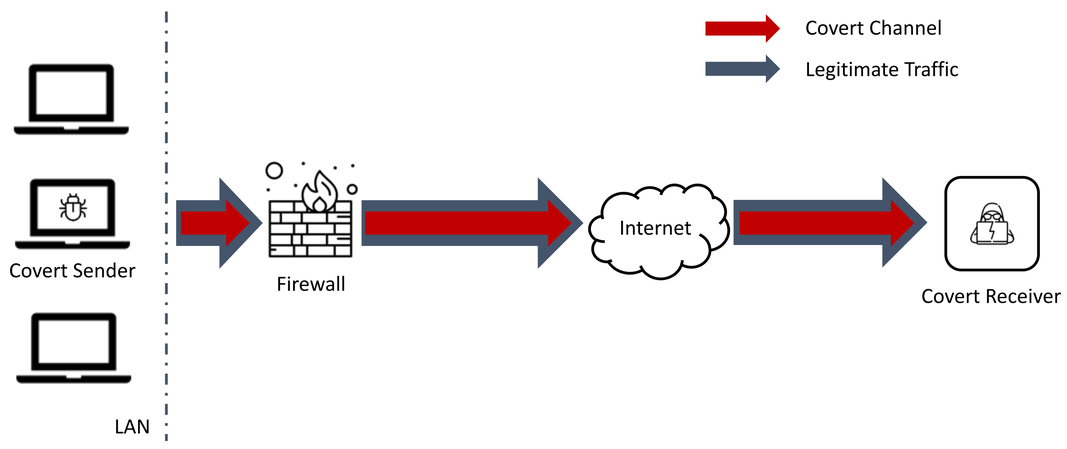

Figure 1: Attack model.

To protect the network infrastructure, security mechanisms inspect both the legitimate and malicious traffic. Unfortunately, the carrier-dependent nature of covert channels and the impossibility of known in advance where the data is hidden make writing detection rules a complicated task. Moreover, to avoid bottlenecks, security mechanisms are often event-based, thus they do not consider per-packet granularity. This is not sufficient to handle all the possible protocol-hiding combinations. For example, to detect a covert channel that uses inter-packet time to encode the secret information, it is necessary to have the entire evolution of the timing statistic.

Tools

Unfortunately, the literature does not offer comprehensive solutions to test security tools against hidden communications. Therefore, to produce realistic network samples and to assess several protocols as well as a wide spectrum of hiding mechanisms, we developed the following tools:

- **IPv6CC** [2]: a framework that allows to implement various network covert channels targeting the IPv6 protocol;
- **pcapStego** [3]: a tool for creating network covert channels starting from *.pcap* traces. The protocols supported are ICMP, ICMPv6, IPv4, and IPv6;
- **TLSCC** [4]: a suite of different covert channels targeting the Transport Layer Security (TLS) protocol.

We evaluated the security capabilities of popular network security tools when dealing with network covert channels. In particular, we investigated:

- **Snort**: it is a rule-based security tool able to identify network malicious activities;
- **Zeek**: it is a scriptable network security monitoring tool that supports investigations on malicious activities;
- **Suricata**: it is a high performance IDS to quickly identify and mitigate sophisticated attacks.

Security Assessment

At first, we tested covert channels implemented via IPv6CC. Results showcased that Zeek is completely insensitive to hidden data transfers targeting IPv6, whereas alerts raised by Suricata and Snort can be considered "noise". Secondly, we extended the analysis by considering the TLS, ICMPv4/v6, TCP/UDP, IPv4 and IPv6 protocols, by using TLSCC and pcapStego. Again, all the out-of-the-box security tools did not detect the presence of any covert channel. This result suggests that these tools can not be considered effective or sufficient for covert channels detection, without further configurations.

Figure 2: Out-of-the-box security tools do not detect covert channels.

Conclusions and Forthcoming Research

In this poster, we showcased how Snort, Zeek and Suricata, configured with a minimal set of rules, are not able to detect the presence of covert channels targeting different network protocols. Moreover, the lack of tools to test security systems against network covert channels required to develop specific solutions such as *IPv6CC* and *pcapStego*.

Future works aim at:

- extending the overall analysis by considering ad-hoc sets of rules, e.g., BroCCaDe for Zeek, as well as different network monitoring tools;
- evaluating covert channels targeting different protocols, e.g., DNS, HTTP, or MQTT;
- investigating the usage of other detection mechanisms, e.g., the extended Berkeley Packet Filter.

References

[1] Wojciech Mazurczyk and Luca Caviglione. Information Hiding as a Challenge for Malware Detection. *IEEE Security & Privacy*, 2(13):89–93, 2015.

[2] Luca Caviglione, Andreas Schaffhauser, Marco Zuppelli, and Wojciech Mazurczyk. IPv6CC: IPv6 Covert Channels for Testing Networks Against Stegomalware and Data Exfiltration. *SoftwareX*, 17:100975, 2022.

[3] Marco Zuppelli and Luca Caviglione. pcapStego: A Tool for Generating Traffic Traces for Experimenting with Network Covert Channels. In *The 16th International Conference on Availability, Reliability and Security*, pages 1–8, 2021.

[4] Corinna Heinz, Marco Zuppelli, and Luca Caviglione. Covert Channels in Transport Layer Security: Performance and Security Assessment. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, 12(4):22–36, 2021.

GitHub Repositories:
https://github.com/Ocram95/pcap_injector
https://github.com/Ocram95/IPv6CC_SoftwareX

CONTACTS

Marco Zuppelli
marco.zuppelli@ge.imati.cnr.it

CSW