Towards Deriving Insights into Data Hiding Methods Using Pattern-based Approach

Wojciech Mazurczyk
Warsaw University of Technology
Warsaw, Poland
wmazurczyk@tele.pw.edu.pl

Steffen Wendzel
Worms University of Applied Science/
Fraunhofer FKIE
Worms/Bonn, Germany
wendzel@hs-worms.de

Krzysztof Cabaj
Warsaw University of Technology
Warsaw, Poland
kcabaj@ii.pw.edu.pl

ABSTRACT

In network information hiding, *hiding patterns* are used to describe hiding methods and their taxonomy. In this paper, we analyze the current state of hiding patterns and we further improve their taxonomy. In order to more thoroughly characterize and understand data hiding methods applied to communication networks we propose to distinguish between sender-side and receiver-side patterns. Additionally, we show how information hiding patterns can be utilized to conveniently describe the realization of the distributed network covert channels.

CCS CONCEPTS

• **Security and privacy → Network security**; *Distributed systems security*; *Information flow control*; Pseudonymity, anonymity and untraceability; • **Social and professional topics** → *Computer crime*;

KEYWORDS

information hiding patterns, network steganography, covert channels; network security; taxonomies; information hiding

ACM Reference Format:
Wojciech Mazurczyk, Steffen Wendzel, and Krzysztof Cabaj. 2018. Towards Deriving Insights into Data Hiding Methods Using Pattern-based Approach. In *ARES 2018: International Conference on Availability, Reliability and Security, August 27–30, 2018, Hamburg, Germany*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3230833.3233261

1 INTRODUCTION

Network covert channels belong to the research domain of *network information hiding* [15]. *Network covert channels* are stealthy, unforeseen communication channels in computer networks. These channels are increasingly used by cybercriminals, e.g. to allow a covert transfer of malware data. However, they can be also used for legitimate purposes, such as communicating illicit information under Internet censorship.

Hiding patterns are descriptions of hiding methods for network covert channels. Because of their abstract nature, each hiding pattern serves an umbrella for numerous hiding methods. For instance, hiding data in the least significant bit (LSB) of the Hop Limit field in IPv6 can be represented by the same pattern as modifying the LSB of the Time to Live field in IPv4. In addition to describing hiding methods, patterns can also form taxonomies and have predefined, searchable and comparable attributes, making them an advantageous tool over existing taxonomy approaches.

Hiding patterns have originally been proposed by Wendzel et al. in [22]. The authors also presented a novel taxonomy of hiding patterns in their article. Later, the taxonomy and patterns were updated and extended by Mazurczyk et al. in [15]. There are also publications that discuss whether a new hiding method can represent a new or an existing pattern [20] and there is moreover work that describes the way in which hiding methods should be described (in the context of patterns) [19].

In this work, we analyze the key aspects of the hiding patterns and the current state of the taxonomy in the domain. However, the main contributions of this paper are that we show how this concept can be further extended by modifying the pattern-analysis process and extending the current taxonomy with new patterns. By taking into account more details on the hiding method's inner workings we hope that the resulting pattern categorization will contribute to a better understanding of the nature of network covert channels. Moreover, we also introduce and describe a pattern-based classification of *distributed* network covert channels.

The rest of this paper is structured as follows. Section 2 introduces fundamentals and related work on hiding patterns. We discuss limitations of the current patterns approach in Section 3. Section 4 introduces our improved taxonomy, a process for pattern-analysis as well as new patterns dedicated to the payload field and our pattern-based categorization of distributed network covert channels. Finally, Section 5 concludes our work and provides an outlook on future research directions.

2 FUNDAMENTALS

To aid the understanding of information hiding methods, an analysis of the existing network covert channels and corresponding protocols should be performed. Patterns provide an abstract and hierarchical view on these methods and their utilization in combination with network protocols.

As a starting point, we utilize the work by Wendzel et al. [22] on network information hiding patterns. In this work, the authors introduce a classification of network hiding techniques into so-called

ARES 2018, August 27–30, 2018, Hamburg, Germany

ACM ISBN 978-1-4503-6448-5/18/08.
https://doi.org/10.1145/3230833.3233261

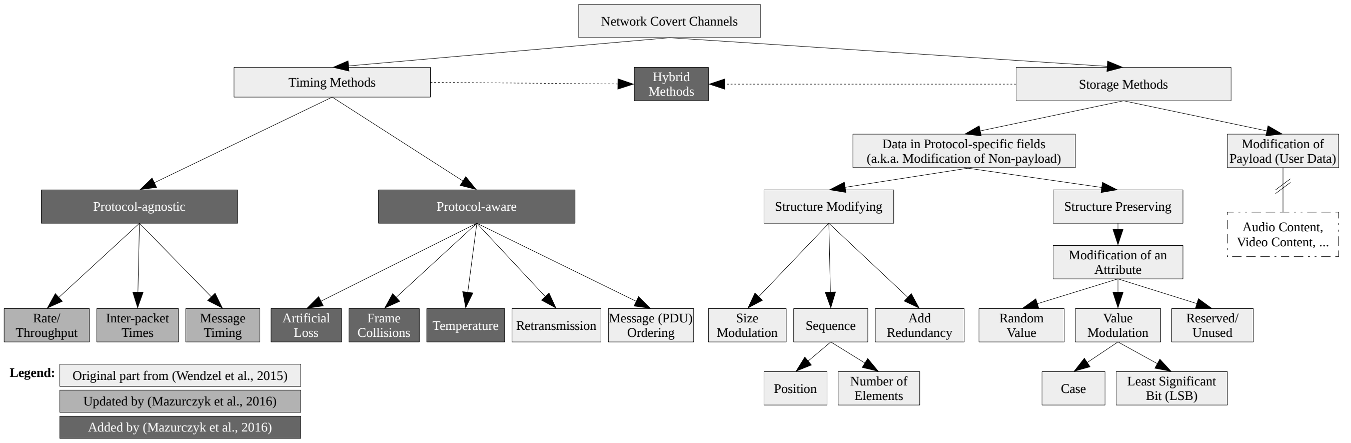


Figure 1: Information hiding patterns and their hierarchy introduced in [22] and updated in [15].

hiding patterns with the aim to potentially develop countermeasures for these patterns. In this perspective, information hiding patterns are defined as abstract descriptions of how to solve a problem (data hiding) in a given context (communication protocols). As patterns can be derived from other patterns, they can form hierarchies. Each hiding pattern is a unified and generic description of a particular family of hiding methods. Patterns must be described in a pre-defined format and require certain additional properties, such as at least three known occurrences of a pattern – cf. [22] for details. In [22] and [19], Wendzel et al. evaluated more than 130 existing network covert channel techniques from past decades and extracted abstract patterns from these techniques. It turned out that authors were able to represent all techniques by (only) 11 patterns, which were arranged in a hierarchical catalog described using *Pattern Language Markup Language* (PLML). While later work in [15] modified and extended their patterns, the core part of the hierarchy and several patterns remained (colored in white and light-gray in Fig. 1). Later modifications and extensions by [15] are colored in darker gray in Fig. 1. The latest description of all patterns shown in this figure is presented briefly in Table 1.

As it can be seen in Table 1, a hiding pattern's description is written in an abstract manner so that one pattern can be used to describe multiple hiding techniques at the same time. For instance, "modulate the least significant bits of a protocol field" is a very brief description of many published hiding methods which utilize the least significant bits of fields in arbitrary network protocols.

The above-mentioned classification is carrier-oriented and a "carrier" is defined as one or more overt traffic flows that pass between the covert sender and the covert receiver, consisting of protocol data units (PDUs, e.g. frames or packets). Typically, the carrier can be multi-dimensional, i.e. it offers many opportunities "places" or "events" for hiding data (called sub-carriers). As in other network covert channel categorizations the two main groups of methods are (Fig. 1):

- *storage methods*: a class of network steganography methods that modify the "places" (sub-carriers) in a carrier to create a storage covert channel. These techniques hide information by modifying e.g. protocol fields, such as unused bits of a header.
- *timing methods*: a class of network steganography methods that modify the timing of "events" of a carrier to create a covert channel. These techniques hide information, e.g. in the timing of protocol messages or packets.

Some important changes have been introduced in [15] when compared with original categorization from [22]. These include:

- defining 14 patterns (8 timing patterns and 6 storage patterns), compared to 11 patterns (4 timing and 7 storage) proposed originally. Note that the increased number of hiding patterns is mainly caused due to adding new layer of classification in [15] for timing patterns which have been divided into "protocol agnostic" or "protocol aware" groups.
- the pattern 'PDU Corruption/Loss Pattern' has been removed from the storage patterns and instead the 'Artificial Loss' pattern which full name is 'Artificial Message/Packet Loss' and the 'Frame Collision' pattern have been added to the list of timing patterns.
- A few patterns have been slightly modified/renamed.

The paper [22] introduced also several other concepts which explain suitably some network covert channels' phenomena, i.e. pattern variation, pattern combination, and pattern hopping.

First, *pattern variation* is a transformation-like approach for covert channels. The utilized network protocol is defined as the pattern's context. Therefore, a pattern's application can change from one network protocol to another – without redesigning the most important aspects and inner workings of the hiding technique itself. Next, *pattern combination* allows the use of multiple patterns at the same time (within the same carrier, e.g. by modifying many subcarriers at once). This is typically performed to increase available steganographic bandwidth – thus in short it is a parallel utilization of multiple network covert channels simultaneously. Finally, *pattern hopping* varies the use of patterns over time – usually it is applied in order to increase stealthiness. This can be briefly summarized as a sequential utilization of various network covert channels in time using different (sub-)carriers.

Table 1: Information hiding patterns as introduced in [22] and updated in [15].

Pattern Name	Pattern Description
Rate/Throughput	The covert channel sender alters the data rate of traffic from itself or a third party to the covert channel receiver.
Inter-packet Times	The covert channel alters timing intervals between network PDUs (interarrival times) to encode hidden data.
Message Timing	Hidden data is encoded in the timing of message sequences, e.g. acknowledging every n'th received packet or sending commands m times.
Artificial Loss	The covert channel signals hidden information via artificial loss of transmitted messages (PDUs).
Frame Collisions	The sender causes artificial frame collisions to signal hidden information.
Temperature	The sender influences a third-party node's CPU temperature, e.g. using burst traffic. This influences the node's clock skew. The clock skew can then be interpreted by the covert receiver by interacting with the node.
Retransmission	A covert channel retransmits previously sent or received PDUs.
Message Ordering	The covert channel encodes data using a synthetic PDU order for a given number of PDUs flowing between covert sender and receiver.
Size Modulation	The covert channel uses the size of a header element or a PDU to encode a hidden message.
Sequence Modulation	The covert channel alters the sequence of header/PDU elements to encode hidden information. This pattern divides further into: P2.a. Position and P2.b. Number of Elements patterns.
Add Redundancy	The covert channel creates new space within a given header element or within a PDU in which to hide data.
Random Value	The covert channel embeds hidden data in a header element containing a (pseudo-)random value.
Value Modulation	The covert channel selects one of the n values that a header element can contain to encode a hidden message. This pattern divides further into: P6.a. Case Pattern and P6.b. Least Significant Bit (LSB) patterns.
Reserved/Unused	The covert channel encodes hidden data into a reserved or unused header/PDU element.

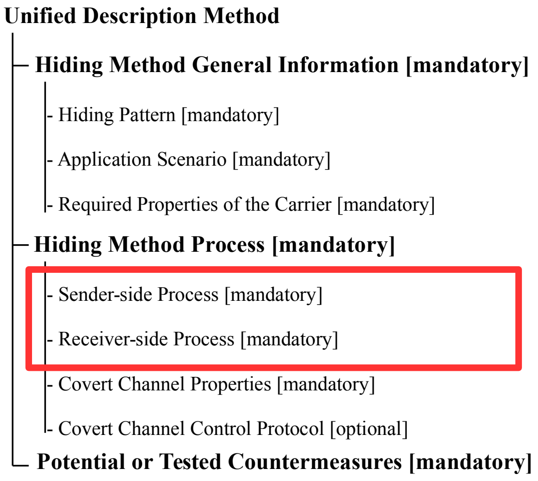

Unified Description Method
Hiding Method General Information [mandatory]
- Hiding Pattern [mandatory]
- Application Scenario [mandatory]
- Required Properties of the Carrier [mandatory]
Hiding Method Process [mandatory]
- Sender-side Process [mandatory]
- Receiver-side Process [mandatory]
- Covert Channel Properties [mandatory]
- Covert Channel Control Protocol [optional]
Potential or Tested Countermeasures [mandatory]

Figure 2: The unified description structure for data hiding methods as introduced in [19].

It must be also noted that in the reminder of this paper we will rely on the unified description for network information hiding methods introduced in [19]. This paper has been the first attempt to standardize the description of network covert channels which is suitable, e.g. to assess their novelty and impact of the method on the state-of-the-art. In [19], the proposed description of data hiding methods is split into three categories: *(i)* general information about the hiding method; *(ii)* description of the hiding process, and *(iii)* potential or tested countermeasures. The first two categories comprise sub-categories and each (sub-)category can be mandatory or optional (Fig. 2).

The category "hiding method general information" consists of a link to existing network hiding patterns. It also includes a discussion of the application scenario and requirements of the carrier. From the perspective of this paper the most important category, i.e. "hiding method process", is split into four parts: the sender-side and the receiver-side description of the hiding method, the details of the covert communication channel, and the description of an associated covert channel control protocol (if applicable). The third category discusses both, potential and evaluated countermeasures, including those that detect, limit or prevent the particular hiding method's use. In the following we will reference to the fragments of this unified description when it comes to the pattern categorization.

3 ANALYSIS OF THE EXISTING TAXONOMY

Our analysis has shown that the current information hiding patterns approach can be further extended to include the following aspects:

- *Incorporate More Details on Data Hiding Methods:* The key criterion of the current pattern taxonomy for deciding which pattern an analyzed method represents is to determine how the secret data is encoded. Thus, due to this it is omitting some details on how the data hiding method works (from the sender-side and receiver-side process – this will be shown further in the next sections). This "flattens" the description of the inner workings of the data hiding methods and thus may prevent that *all* details of a hiding method are considered. A more thorough patterns grouping is desired to more accurately categorize existing network steganography methods.
- *Support Hybrid Patterns:* For some cases it is difficult to assess whether the analyzed method is storage, timing or hybrid – a clearer distinction and unambiguous formula to deduce this is desirable.

- *Multi-Packet and -Flow Characteristics Support:* The current categorization makes no clear distinction between hiding methods that are focusing on a single packet or multiple packets. Also, there is no clear distinction between single- and multi-flow methods. For example, consider a covert channel that modulates IPv4 ToS values in such a way that the sequence of ToS values from the consecutive packets is interpreted as a single secret data bit – currently such a method does not match any hidden data pattern. Moreover, some hiding methods such as [10] utilize multiple flows. It is thus beneficial to make the original pattern descriptions more generic, i.e. less dependent on specific units such as PDUs or packets.
- *Coverage of Sophisticated Hiding Methods:* It is not exactly clear whether recent, more advanced network steganography concepts like inter-protocol steganography [9], protocol switching covert channels [21], multilevel steganography [5], adaptive covert communication [23], etc. can be accurately expressed using current pattern categorization. Pattern combination, pattern hopping and pattern variation are means to represent them, but not to the full extent.
- *Influence on Payload:* In the original design decision of the pattern-based approach, arbitrary content, e.g. digital files, were considered as part of digital media steganography instead of network information hiding. However, in some cases, such as in VoIP steganography, where there are data hiding methods that affect the payload field, it can be helpful to have a taxonomy that covers also the transmitted payload. In principle the patterns should be analogous as they too adhere to the storage group.
- *Distinction Between Secret Data Embedding and Transfer:* It is also worth to emphasize that from the pattern-based countermeasures perspective it is more important to know which pattern represents the covert technique *within* the communication channel. It must be noted that in particular the information hiding patterns used at the sender-side process to embed secret data may not exactly represent themselves in the same while traversing within the hidden data carrier through the communication network.
- *Embrace PDU Corruption Pattern:* As mentioned, in [22] 11 (4 timing and 7 storage) patterns have been defined while in [15] there are 14 (8 timing and 6 storage) patterns. However, the pattern 'PDU Corruption/Loss' has been removed from the storage patterns group by [15]. In fact, it is our belief that it is beneficial that the 'Artificial Message/Packet Loss' pattern has been added into timing patterns but still the 'PDU Corruption' pattern should be considered in storage scenarios.

Based on the above-mentioned points, we describe how we envision enhancements to the current information hiding patterns concept in the next section.

4 EXTENSION AND MODIFICATION OF THE PATTERNS APPROACH

In this section, we present the proposed modification for the original information hiding patterns concept which can help in deriving further insights into understanding the nature of various types of network covert channel techniques. More specifically, in subsection 4.1 we propose how the original pattern approach can be extended in order to include the sender-side and receiver-side processes which influences both pattern creation process and covert techniques categorization. Next, in subsection 4.2 we propose new patterns applicable to the payload field. Finally, in subsection 4.3 we discuss the *distributed* network covert channels and how the information hiding patterns concept can be used to conveniently describe them.

4.1 A New Process to Analyze the Details of Pattern-Application

Considering the arguments from Section 3, we propose an approach based on [20], which describes how to determine the novelty of a new hiding technique and whether a hiding technique actually represents a *new* pattern, or not. Instead, our goal is to gain additional insights into the inner-workings of the data hiding method, i.e. we do not *replace* the original approach.

In the current categorization, authors of a new data hiding technique first describe their technique, e.g. informally or using [19]. Then, based on how the secret data is *embedded* one pattern is selected that represents the hiding method. Therefore, authors first decide whether the hiding method is storage or timing, then, whether it is protocol-aware/agnostic (timing channel) or header structure preserving/modifying (storage channel). If a hiding method does *not* fit into the current pattern representation, it is considered a *new* pattern which can be added to the taxonomy. The related decision-making process can be found in [20].

We propose a similar but modified version of this approach. However, as mentioned, our approach targets a different goal, namely to derive *more insights* related to the information hiding method itself. It must be noted that we do not focus only on how the secret data for a certain data hiding method is embedded (which is only a part of the sender-side process) but instead we want to detail both the complete sender- and receiver-side processes and represent them with patterns (and for this purpose, we "borrow" the already existing patterns.

In our proposed approach, the known hiding patterns of existing publications and websites, e.g. [15, 22] or https://ih-patterns.blogspot.com, which are tagged as storage or timing patterns, are taken into account. Then for the hiding method that needs to be described using the network covert channel patterns approach, the corresponding patterns for both, sender- and receiver-side processes are selected. Finally, based on the result and depending on what types of patterns have been assigned to the method, the *method itself* is concluded as a storage, a timing or a hybrid method – this selection process is explained in the details below.

The described improved approach which aims to derive more insights from the data hiding methods using pattern approach allows to repaint the categorization from Fig. 1. However it must be noted that in the modified approach we categorize *network covert channel patterns* and not data hiding *methods.* Thus, we start the derived classification from the network covert channel patterns which are then divided into timing and storage ones (Fig. 3). Afterwards, each of the methods that needs to be evaluated is assigned with at least

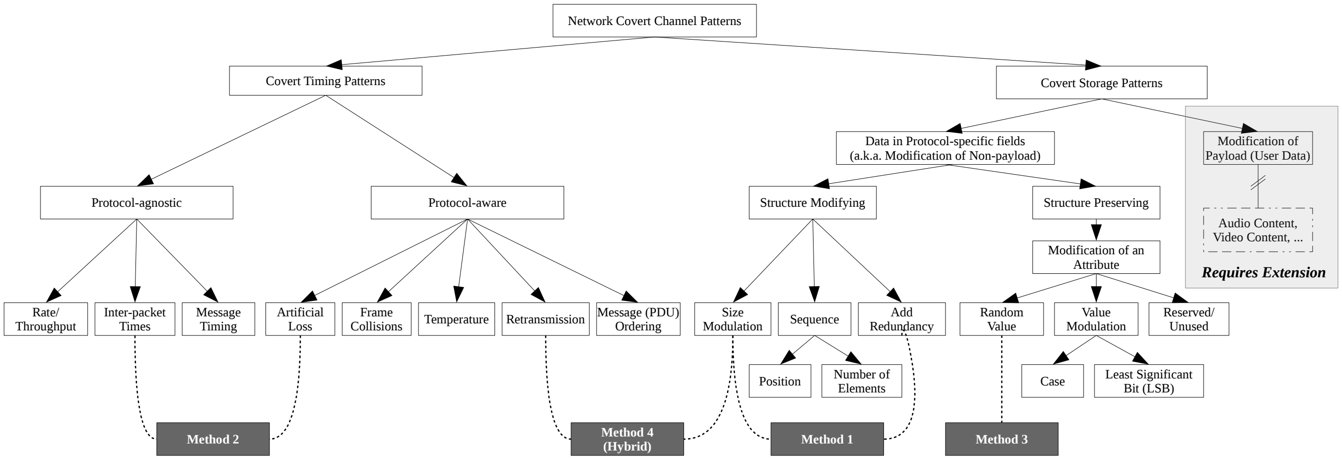


Figure 3: Improved aspects of the existing pattern-based taxonomy.

one or more patterns to its sender- and receiver-side processes separately (for each side at least one or alternatively more patterns must be selected).

It must be also noted that using this approach it may be possible to evaluate in greater detail which patterns are most often used jointly only at the sender-side process (as more than one pattern can be assigned) or only at the receiver-side process, or finally which patterns typically coexist at the sender-side and the receiver-side processes. This can be achieved by performing a thorough analysis of network covert channels defined in the literature (however, due to space limitation it will be not part of this paper). In result of such an analysis this can lead to the identification of potential relationships between defined patterns, i.e. whether for some of them it is "easier" to coexist with other patterns within the data hiding method (as in the case of the extended approach the sender and the receiver processes can be investigated separately or jointly).

But more importantly, it is also possible to investigate whether besides of joint patterns utilization (at the sender-side, receiver-side or both sides), other pattern mixes are also possible. For example, consider Method 4 in the Figure 3. It is characterized by the patterns *Retransmission* and *Size Modulation*, which makes it a hybrid method. However, the question arises whether is would be possible to construct a data hiding method that apart from these two patterns utilizes e.g. *Message (PDU) Ordering* pattern and how this will impact its properties.

In result, new, previously unknown network information hiding methods or improved versions of existing ones can be designed and developed and relationships between the existing patterns can be investigated and determined. It must be noted that using the existing pattern classification it was possible to assign only a single pattern for a certain hiding method which corresponds best with the secret data embedding process. However, in the extended approach (which is different when compared to the original concept) it is possible to:

- assign more patterns to the sender-side process if it is required in order to express to a full extent how the sender-side of the hiding method operates,

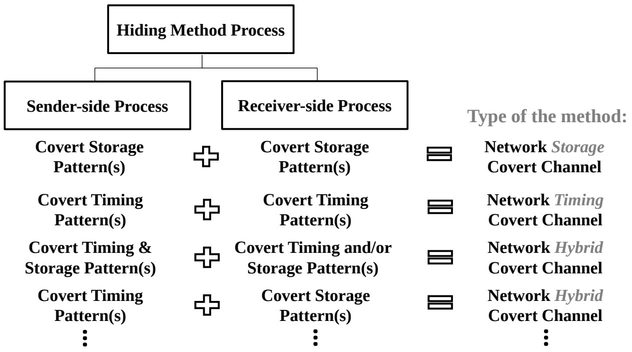
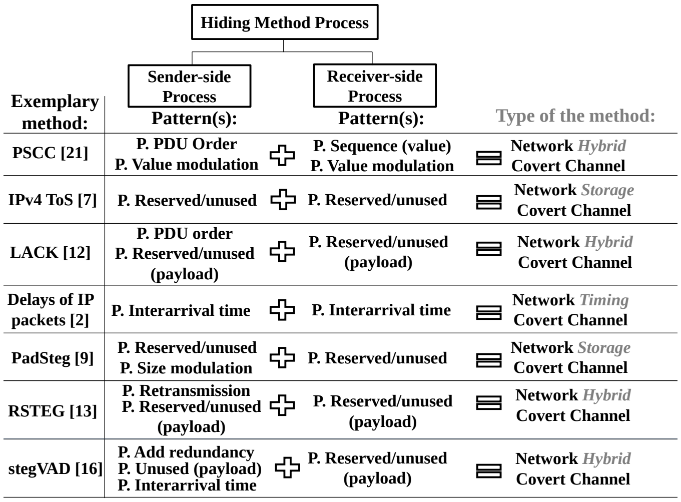

Hiding Method Process
Sender-side Process
Receiver-side Process
Type of the method:
Covert Storage Pattern(s)
Covert Storage Pattern(s)
Network Storage Covert Channel
Covert Timing Pattern(s)
Covert Timing Pattern(s)
Network Timing Covert Channel
Covert Timing & Storage Pattern(s)
Covert Timing and/or Storage Pattern(s)
Network Hybrid Covert Channel
Covert Timing Pattern(s)
Covert Storage Pattern(s)
Network Hybrid Covert Channel

Figure 4: Improved process to decide on the network covert channel type based on the assigned patterns.

- include also the receiver-side process and its corresponding patterns.

Such an approach may not only help to better understand the nature of the network covert channels and their creation process, but it can also provide new insights into how to construct more efficient and effective detection solutions. This can be achieved by designing and developing detection methods, so they precisely will be looking for the specific artifacts related to the representation of the certain patterns in the communication channel (and/or e.g. the presence of their coexistence).

Finally, each method based on the selected patterns for the sender- and for the receiver-side processes is assigned to one element of the group {*storage*, *timing*, *hybrid*}. This is done as illustrated in Fig. 4. In principle, if both the sender- and the receiver-side processes are characterized with homogenic (only storage or only timing) patterns then the method is concluded as storage or timing. If there is heterogeneity across patterns that the method uses, i.e. storage and timing methods are mixed within the sender- and/or the receiver-side processes then it is concluded as a hybrid technique.


Hiding Method Process
Sender-side Process
Receiver-side Process

Exemplary method:	Sender-side Process Pattern(s):		Receiver-side Process Pattern(s):		Type of the method:
PSCC [21]	P. PDU Order P. Value modulation	+	P. Sequence (value) P. Value modulation	=	Network *Hybrid* Covert Channel
IPv4 ToS [7]	P. Reserved/unused	+	P. Reserved/unused	=	Network *Storage* Covert Channel
LACK [12]	P. PDU order P. Reserved/unused (payload)	+	P. Reserved/unused (payload)	=	Network *Hybrid* Covert Channel
Delays of IP packets [2]	P. Interarrival time	+	P. Interarrival time	=	Network *Timing* Covert Channel
PadSteg [9]	P. Reserved/unused P. Size modulation	+	P. Reserved/unused	=	Network *Storage* Covert Channel
RSTEG [13]	P. Retransmission P. Reserved/unused (payload)	+	P. Reserved/unused (payload)	=	Network *Hybrid* Covert Channel
stegVAD [16]	P. Add redundancy P. Unused (payload) P. Interarrival time	+	P. Reserved/unused (payload)	=	Network *Hybrid* Covert Channel

Figure 5: Classification of the exemplary network covert channels based on the assigned patterns.

To present how the proposed extended patterns' classification approach is functioning for some of the existing network steganography techniques, we have chosen seven different state-of-the-art network covert channels to demonstrate how they fit into our categorization (Fig. 5). For example, for a simple network covert channel which in order to conceal data utilizes Type of Service field from the IPv4 header [7], the sender- as well as receiver-side processes use the same pattern, i.e. *Reserved/Unused*, thus as both processes are assigned with the storage pattern then the method is concluded as storage. For the work related to modifying delays between the consecutive packets within the data stream [2] for both sender- and receiver-side processes the pattern *Inter-arrival time* is an obvious choice thus this technique is deemed as timing method. However, when we consider a more complex method like LACK (*Lost Audio Packets Steganography*) [12] then the situation is a bit different. As LACK operates by using intentionally delayed voice packets and replacing the original payload of these packets with secret data thus at the sender-side process two patterns must be selected – one storage (*Reserved/Unused*) and one timing (*PDU Order*), whereas when considering the receiver-side process the chosen pattern is only storage one (*Reserved/Unused*) – as at the covert receiver every incoming packet's payload, regardless of its order, is probed for the existence of the hash which will indicate presence of secret data. Therefore, the method is concluded to be hybrid. It is worth emphasizing that if we consider the original pattern approach (which as mentioned relied only on assigning pattern(s) based on how/where secret data is embedded) then LACK method would be only characterized by the storage *Reserved/Unused* pattern. This proves that the extended pattern approach proposed in this paper allows to characterize the data hiding methods in greater detail by including more information on inner workings of the information hiding technique.

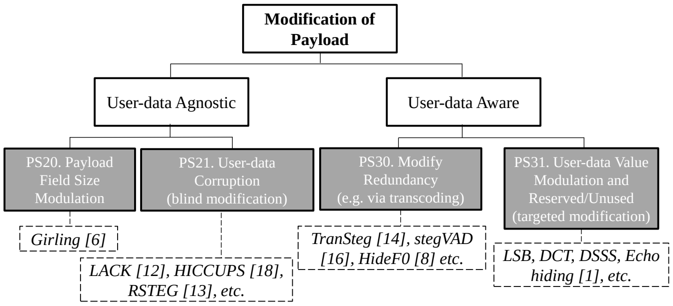

Modification of Payload
User-data Agnostic
User-data Aware
PS20. Payload Field Size Modulation
PS21. User-data Corruption (blind modification)
PS30. Modify Redundancy (e.g. via transcoding)
PS31. User-data Value Modulation and Reserved/Unused (targeted modification)
Girling [6]
LACK [12], HICCUPS [18], RSTEG [13], etc.
TranSteg [14], stegVAD [16], HideF0 [8] etc.
LSB, DCT, DSSS, Echo hiding [1], etc.

Figure 6: Classification of the network covert storage channels for the payload field and the corresponding patterns.

4.2 Introduction of Additional Patterns

As already mentioned, the current pattern-based categorization of [15, 22] makes a distinction between patterns applied to user-data (within the payload field) and protocol specific data (control information: headers, padding, etc.). In principle, all these patterns adhere to the storage group, i.e. modification of the certain "locations" of the carrier. However, in the original publications on hiding patterns, this distinction was made based on the idea of Fisk et al. [3] to separate structured (machine-readable) content from non-structured (human-readable) content, such as images. This means that in several cases similar rules apply to modify these fields (because structured data follows rules, e.g. protocol headers are built similarly to formal grammar) and to the data that they store. Obviously the most significant difference lays in the dissimilarities between the control information carried within protocol headers/padding and user-data transferred within the payload field. Thus, to fill this gap and by considering current research efforts in this area, we propose to extend the current taxonomy as shown in Fig. 6.

Network covert channels that modify the payload field and its content have been divided based on whether the characteristic of user-data is taken into account into: *(i)* user-data agnostic and *(ii)* user-data aware. In each of the two groups two patterns have been identified, which we describe in the same way as the patterns were originally described in [22] using a subset of the *Pattern Language Markup Language's* (PLML) attributes:

PS20. Payload Field Size Modulation

Illustration: This pattern uses a size of the payload field of a flow's PDUs/messages to encode the hidden message. This pattern is a variant (child) of the pattern *P1. Size Modulation* of [22] which has been already defined for the modification of the non-payload branch of storage methods (confirm Fig. 1).
References: PS1. Size Modulation
Context: Network Covert Channel Patterns → Covert Storage Channel Patterns → Modification of Payload → User-data Agnostic
Evidence:
1. Modulate the size of the data block field in Ethernet frames [6].
2. Any other method that modulates the size of the payload field in any network protocol.

PS21. User-data Corruption
Illustration: This pattern is related to the cases when steganographic methods do not take into account what kind of user-data is carried within a payload field and/or what its characteristic is (blind modification). It can be applied to single PDUs or to multiple PDUs (a flow). This typically happens if parts of (or the whole) user-data is replaced with secret bits and thus the user-data is corrupted/lost. This pattern is similar to the pattern *PDU Corruption* defined in the original pattern categorization of [22].
Context: Network Covert Channel Patterns → Covert Storage Channel Patterns → Modification of Payload → User-data Agnostic
Evidence:
1. Replace the user-generated data in the payload field with secret data in intentionally lost voice packets of the IP telephony call [12].
2. Replace the user-generated data in the payload field with secret data in retransmitted TCP segments [13].
3. Replace the user-generated data in the payload field with secret data in intentionally corrupted IEEE 802.11 frames [18].

PS30. Modify Redundancy
Illustration: This pattern is used when it is possible to exploit the redundancy of the user-data by means of transforming them in such a way that a free space for secret data is obtained (e.g. by means of transcoding). This pattern is a bit similar to the pattern *Add Redundancy* defined in [22] but can also decrease redundancy and is applied to payload instead of meta-data.
Context: Network Covert Channel Patterns → Covert Storage Channel Patterns → Modification of Payload → User-data Aware
Evidence:
1. Compress existing user-data in order to make a space for secret data [14].
2. Transform the VAD-enabled IP telephony voice stream into non-VAD one and fill the gaps using artificially generated RTP packets containing secret data [16].
3. Approximate the F0 parameter of the Speex codec which carries information about the pitch of the speech signal and use the “saved” space for secret data [8].

PS31. User-data Value Modulation and Reserved/Unused
Illustration: Characteristic features of user-data can be utilized to store secret information. This includes applying methods like LSB modification to speech samples or digital images carried within the payload field. Compared with previous patterns this is a targeted modification. This pattern is analogous to the combination of the patterns *Value Modulation* and *Reserved/Unused*, but applied to payload.
Context: Network Covert Channel Patterns → Covert Storage Channel Patterns → Modification of Payload → User-data Aware
Evidence:
1. Encode a stream of information by spreading the encoded data across as much of the frequency spectrum as feasible (e.g. DSSS) [1].
2. Embeds secret data into a carrier audio signal by introducing an echo (a.k.a. echo hiding) [1].
3. Replacing the least significant bit of e.g. each voice sample with secret data (LSB) [1].

As it is visible above, the identified patterns have mostly a number of examples in the state-of-the-art publications (Fig. 6). Every newly defined pattern corresponds to the patterns that have been already defined in the *non-payload* branch of the original classification.

Finally, the complete picture of the extended information hiding patterns classification is illustrated in Fig. 7 and the corresponding descriptions of all defined patterns which include also potential multi-packet/multi-flow characteristics of some data hiding methods are enclosed in Tab. 2.

4.3 Distributed Covert Channel Realization

In [22], authors defined three concepts which can be used to explain suitably some of the existing network covert channels' phenomena, i.e. pattern variation, pattern combination and pattern hopping.

The above-mentioned concepts are especially suitable and important when trying to depict, explain, and analyze the realization of *distributed* network covert channels. We define a *distributed covert channel* as a network covert channel that spreads secret data among multiple flows/protocols/hosts or uses multiple patterns within the same flow or PDU for the hidden data exchange. In contrast, the typical (undistributed) network covert channel is a storage or a timing channel that uses PDUs of a single flow/protocol with only one hiding pattern in order to embed secret data.

In Fig. 8 we have illustrated that these three pattern concepts practically exhaust possibilities for distributed network covert channel realization. While explaining these concepts we apply the terms of *spatial*, *temporal*, and *transform domains* which are “borrowed” from the digital media steganography research area [17] and which helps to described and define them better.

The first group i.e. *pattern combination* is related to the distribution of secret data in a *spatial domain*. This means that many patterns are utilized in parallel for the same hidden data carrier e.g. by modifying many of its sub-carriers or using several carriers at once. This includes the case when the hybrid data hiding methods are used (cf. Fig. 1) as well as the case of simultaneous utilization of multiple network covert channels at once. Consider an example of HTTP traffic (e.g. web browsing) where three separate network covert channels are used simultaneously: one is used for the IPv4 protocol, the next for the TCP protocol, and finally the third is applied to HTTP. Pattern combination applies also to the case when, e.g. three separate connections are used for hidden data purposes and in each connection a separate network hiding pattern is utilized at the same time (e.g. IPv4-based in the first connection, TCP-based in the second, and HTTP-based in the last one). Typically such an approach is used in order to increase the overall steganographic bandwidth.

The second group of distributed covert channels realization is *pattern hopping* which allows to spread secret data in the temporal domain (time). In a nutshell it means that different patterns' utilization varies over time and thus they are applied sequentially for various (sub-)carriers. Usually, such an approach helps to improve the stealthiness of the covert data exchange as in order to detect it more “locations” must be monitored by the warden. An example of pattern hopping is the tool *PHCCT*. PHCCT implements a so-called *protocol hopping covert channel* that distributes data over different

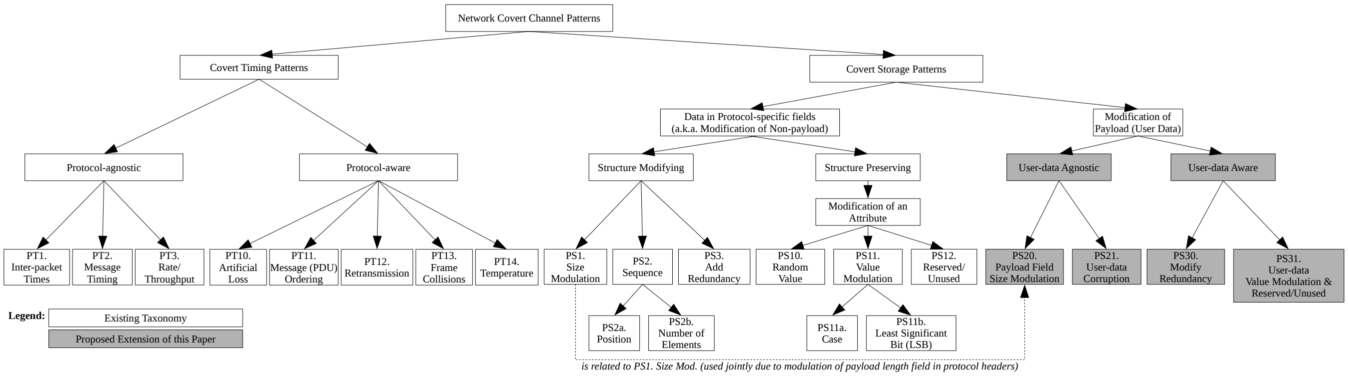


Figure 7: Classification of network covert channel patterns.

Table 2: Descriptions of hiding patterns in our improved and extended taxonomy.

Pattern Name	Pattern Description
PT1. Inter-packet Times	The covert channel alters timing intervals between network messages of a flow (interarrival times) to encode hidden data.
PT2. Message Timing	Hidden data is encoded in the timing of message sequences within a flow, e.g. acknowledging every *n*'th received message or sending commands *m* times.
PT3. Rate/Throughput	The covert channel sender alters the data rate of a flow from itself or a third party to the covert receiver.
PT10. Artificial Loss	The covert channel signals hidden information via artificial loss of a flow's transmitted messages, e.g. by frame-corruption or message drop.
PT11. Message Ordering	The covert channel encodes data using a synthetic message order in a flow.
PT12. Retransmission	A covert channel retransmits previously sent or received messages of a flow.
PT13. Frame Collisions	The sender causes artificial frame collisions to signal hidden information.
PT14. Temperature	The sender influences a third party node's hardware temperature using traffic of a flow. There must be a technique for the covert receive to measure the temperature (indirectly).
PS1. Size Modulation	The covert channel uses the size of flow metadata (e.g. PDU size or size of a header element) to encode hidden messages.
PS2. Sequence Modulation	The covert channel alters the sequence of flow metadata to encode hidden information. This pattern divides further into: P2.a. Position and P2.b. Number of Elements patterns.
PS3. Add Redundancy	The covert channel embeds redundant metadata (e.g. by adding an unused IP option) in which data is hidden into a flow. Note that in comparison to PS1, the data is hidden in the redundant data's presence, not in the size of an PDU or header element).
PS10. Random Value	The covert channel embeds hidden data into flow metadata that contains a (pseudo-)random value.
PS11. Value Modulation	The covert channel selects one of the *n* values that a flow's metadata element can contain to encode a hidden message. This pattern divides further into: P11.a. Case Pattern and P11.b. Least Significant Bit (LSB) patterns.
PS12. Reserved/Unused	The covert channel encodes hidden data into a flow's reserved or unused metadata elements.
PS20. Payload Field Size Modulation	The size of the payload in a flow is used to encode hidden information (this is a derivate of PS1 but for the payload since it involves the modification of a PDU's payload length field, i.e. PS1).
PS21. User-data Corruption	The covert channel performs a (blind) insertion of covert data into a flow's payload (similar PT10).
PS30. Modify Redundancy	The covert channel compresses a flow's payload and the resulting free space is used to hide data.
PS31. User-data Value Modulation and Reserved/Unused	The covert channel performs a modification of a flow's payload in a way that is not reflected by PS30 and that does not result in a significantly modified interpretation of the data, e.g. by modifying least significant bits of digital images or hiding data in unused/reserved payload bits.

network protocols [15]. To this end, PHCCT utilizes more than one pattern, namely *Add Redundancy* (embedded in HTTP) and *User-data Corruption* (embedded in FTP-Data).

Finally, the last group of techniques which allows to realize a distributed network covert channel is *pattern variation.* The original idea of pattern variation is that each of the defined patterns is considered in the certain context, i.e. the utilized hidden data carrier

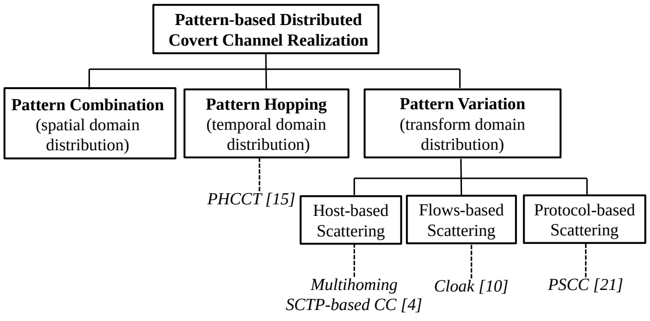

Pattern-based Distributed Covert Channel Realization
Pattern Combination (spatial domain distribution)
Pattern Hopping (temporal domain distribution)
Pattern Variation (transform domain distribution)
PHCCT [15]
Host-based Scattering
Flows-based Scattering
Protocol-based Scattering
Multihoming SCTP-based CC [4]
Cloak [10]
PSCC [21]

Figure 8: Classification of pattern-based distributed network covert channel realization.

(e.g. a network protocol). In our case, we extend this view and define pattern variation in different contexts. In particular, three contexts can be distinguished: *host-based scattering*, *flow-based scattering*, and *protocol-based scattering* which will be described in detail with examples below. In all cases of pattern variation, the *same* pattern is applied to *different* contexts, i.e. its essence does not change.

Host-based scattering requires the covert sender and/or the covert receiver to control more than one physical host or other networking devices. Parts of the secret data are hidden in the legitimate traffic sent from or directed towards different hosts using the same pattern. An example of this kind of distributed covert channel is the SCTP multi-homing-based method (i.e. the host's ability to be visible in the network through more than one IP address) [4]. In such a scenario, each IP address of the covert receiver can be used to represent a single bit of secret data (or a sequence of bits). Then, by modulating the way that packets are addressed and sent secret data can be transferred in a distributed manner.

Next, *Flow-based* scattering takes advantage of the capability to set up multiple flows between two hosts and using them to signal secret data bits in a distributed way while utilizing the same pattern. This can be realized, for example, by dividing secret data into fragments and using a certain information hiding pattern (or several) to send each fragment using one of the available flows. An idea of using many flows for a distributed covert channel is exemplified by the Cloak method [11], which is a timing data hiding technique that encodes secret data bits by uniquely distributing N packets over M TCP flows. Please note that while in the case of *pattern hopping* a utilization of multiple flows is possible as well, *flow-based scattering* serves under the umbrella of pattern variation, i.e. it is required to apply the *same pattern* to *different flows*, and pattern hopping must apply *different patterns*.

Finally, *Protocol-based* scattering applies a pattern to different communication protocols instead of hosts or flows. In contrast to flow-based scattering, it does not necessarily utilize flows of the same protocol but changes the actual protocol (which can generate multiple flows, too). This group is exemplified via *protocol switching covert channels* (PSCC) [21]. These channels assign hidden information to network protocols. For instance, one could link the HTTP protocol to the hidden value "0" and the DNS protocol to the hidden value "1". Then, by sending the packet sequence HTTP, DNS, DNS, HTTP, one would transfer the secret information "0110".

Obviously, there are other possibilities to create distributed network covert channels by developing mixed solutions so that it involves the parallel use of, e.g. pattern hopping and pattern variation or any other fusion of the concepts mentioned above.

5 CONCLUSIONS

We identified limitations of the existing pattern-based taxonomy, most importantly a lack of payload-based hiding patterns and a limited definition of distributed covert channels. For this reason, we extended the list of existing hiding patterns for network covert channels and their related taxonomy. We also extended the description of hybrid/distributed hiding methods and proposed an extension and improvement of the related concepts (especially pattern variation to handle multi-host, multi-flow and multi-protocol techniques).

We hope this work will help to derive new insights into existing and new data hiding techniques.

Future work will be devoted to analyzing relationships between patterns with respect to their joint occurrence in existing methods as well as we will investigate whether any new data hiding methods can be deuced based on the less obvious pattern mixes.

ACKNOWLEDGMENTS

Wojciech Mazurczyk and Krzysztof Cabaj are supported by the Air Force Office of Scientific Research under award number FA9550-17-1-0254. The supported project is named CoCoDe (Covert Communication Detection).

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Air Force.

REFERENCES

[1] W. Bender, D. Gruhl, N. Morimoto, and A. Lu. 1996. Techniques for data hiding. *IBM Systems Journal* 35, 3.4 (1996), 313–336. https://doi.org/10.1147/sj.353.0313

[2] V. Berk, A. Giani, and G. Cybenko. 2005. *Detection of Covert Channel Encoding in Network Packet Delays.* Technical Report TR2005-536. Department of Computer Science, Dartmouth College. http://www.ists.dartmouth.edu/library/149.pdf http://www.ists.dartmouth.edu/library/149.pdf.

[3] G. Fisk, M. Fisk, C. Papadopoulos, and J. Neil. 2003. Eliminating steganography in Internet traffic with active wardens. In *Proc. Revised Papers from the 5th International Workshop on Information Hiding*. 18–35.

[4] Wojciech Fraczek, Wojciech Mazurczyk, and Krzysztof Szczypiorski. 2012. Hiding Information in a Stream Control Transmission Protocol. *Comput. Commun.* 35, 2 (Jan. 2012), 159–169. https://doi.org/10.1016/j.comcom.2011.08.009

[5] W. Fraczek, W. Mazurczyk, and K. Szczypiorski. 2012. Multilevel Steganography: Improving Hidden Communication in Networks. *Journal of Universal Computer Science* 18, 14 (jul 2012), 1967–1986.

[6] C. G. Girling. 1987. Covert Channels in LAN's. *IEEE Transactions on Software Engineering* 13, 2 (1987), 292–296.

[7] Theodore G. Handel and Maxwell T. Sandford. 1996. Hiding data in the OSI network model. In *Information Hiding*, Ross Anderson (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg, 23–38.

[8] Artur Janicki. 2016. Pitch-based Steganography for Speex Voice Codec. *Security and Communication Networks* 9, 15 (2016), 2923–2933. https://doi.org/10.1002/sec.1428

[9] B. Jankowski, W. Mazurczyk, and K. Szczypiorski. 2013. PadSteg: introducing inter-protocol steganography. *Telecommunication Systems* 52, 2 (01 Feb 2013), 1101–1111. https://doi.org/10.1007/s11235-011-9616-z

[10] X. Luo, E. W. W. Chan, and R. K. C. Chang. 2007. Cloak: A Ten-Fold Way for Reliable Covert Communications. In *Computer Security – ESORICS 2007*, Joachim Biskup and Javier López (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 283–298.

[11] Xiapu Luo, Edmond W. W. Chan, and Rocky K. C. Chang. 2007. Cloak: A Ten-Fold Way for Reliable Covert Communications. In *Computer Security – ESORICS*

2007, Joachim Biskup and Javier López (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 283–298.

[12] Wojciech Mazurczyk and Józef Lubacz. 2010. LACK—a VoIP steganographic method. *Telecommunication Systems* 45, 2 (01 Oct 2010), 153–163. https://doi.org/10.1007/s11235-009-9245-y

[13] W. Mazurczyk, M. Smolarczyk, and K. Szczypiorski. 2011. Retransmission steganography and its detection. *Soft Computing* 15, 3 (2011), 505–515. https://doi.org/10.1007/s00500-009-0530-1

[14] Wojciech Mazurczyk, PawełSzaga, and Krzysztof Szczypiorski. 2014. Using Transcoding for Hidden Communication in IP Telephony. *Multimedia Tools Appl.* 70, 3 (June 2014), 2139–2165. https://doi.org/10.1007/s11042-012-1224-8

[15] W. Mazurczyk, S. Wendzel, S. Zander, A. Houmansadr, and K. Szczypiorski. 2016. *Information Hiding in Communication Networks: Fundamentals, Mechanisms, Applications, and Countermeasures.* Wiley-IEEE.

[16] Sabine S. Schmidt, Wojciech Mazurczyk, Jörg Keller, and Luca Caviglione. 2017. A New Data-Hiding Approach for IP Telephony Applications with Silence Suppression. In *Proceedings of the 12th International Conference on Availability, Reliability and Security (ARES '17).* ACM, New York, NY, USA, Article 83, 6 pages. https://doi.org/10.1145/3098954.3106066

[17] Mansi S. Subhedar and Vijay H. Mankar. 2014. Current status and key issues in image steganography: A survey. *Computer Science Review* 13-14 (2014), 95 – 113. https://doi.org/10.1016/j.cosrev.2014.09.001

[18] Krzysztof Szczypiorski. 2012. A performance analysis of HICCUPS—a steganographic system for WLAN. *Telecommunication Systems* 49, 2 (01 Feb 2012), 255–259. https://doi.org/10.1007/s11235-010-9363-6

[19] S. Wendzel, W. Mazurczyk, and S. Zander. 2016. Unified Description for Network Information Hiding Methods. *Journal of Universal Computer Science* 22, 11 (nov 2016), 1456–1486.

[20] S. Wendzel and C. Palmer. 2015. Creativity in Mind: Evaluating and Maintaining Advances in Network Steganographic Research. *Journal of Universal Computer Science* 21, 12 (2015), 1684–1705.

[21] S. Wendzel and S. Zander. 2012. Detecting Protocol Switching Covert Channels. In *37th IEEE Conf. on Local Computer Networks.* 280–283.

[22] S. Wendzel, S. Zander, B. Fechner, and C. Herdin. 2015. Pattern-based Survey and Categorization of Network Covert Channel Techniques. *Computing Surveys (CSUR)* 47, 3 (2015).

[23] F. V. Yarochkin, S.-Y. Dai, C.-H. Lin, and Y. Huang. 2008. Towards Adaptive Covert Communication System. In *Proc. Pacific Rim International Symposium on Dependable Computing (PRDC).* 153–159.