Hiding Data in a Switched Network

Aleksandra Mileva* and Jordan Tikvesanski
University "Goce Delcev"
Stip, Republic of N. Macedonia
{aleksandra.mileva, jordan.tikvesanski}@ugd.edu.mk

Received: May 29, 2021; Accepted: August 22, 2022; Published: September 30, 2022

Abstract

This paper presents two novel methods for hiding data in Cisco switches as intermediate innocent devices, from the entire VTP domain. New steganographic methods affect the switches in the distribution and access layer of the three - tier hierarchical network model. They are using a combination of Switched spoofing VLAN attack (a kind of VLAN hopping), and a version of "VTP bomb" attack, to trigger the cover storage and transfer. An experimental testbed was created for a proof-of-concept and a steganographic analysis of the newly created covert channels is performed. At the end, proper countermeasures are suggested.

Keywords: VTP, Covert channels, VLAN Trunking Protocol, Network steganography

1 Introduction

Many organizations and enterprises have large networks designed by using three - tier hierarchical network model. This model is built of an *access layer* of hubs, bridges, switches, or routers, connected to end devices (e.g., servers, printers, computers), a *distribution layer* of mid-tier routers and switches implementing forwarding and routing decisions, and a *core layer* of high-end routers and switches that connect the distribution network devices between themselves and to outside services such as the Internet. This model heavily deploys the concept of Virtual Local Area Network (VLAN) as a logical group of hosts that belong to the same broadcast domain, while their physical location can be on different physical LANs.

Network steganography as an information hiding sub-discipline, deals with hidden data transfers over communication networks and their detection. Used steganographic methods usually deploy network protocols and their legitimate network flows for transferring and/or storing the hidden data. This can result in a creation of covert channels between the communicating parties. Many network protocols have been subject of study of network steganography for several decades [1, 2]. However, to our best knowledge, there are no published papers on deploying the entire switched network and the VLAN Trunking Protocol (VTP) for creation of covert channels. The tasks of identification of network covert channels and suggestion of proper countermeasures against them, are very important from the security perspective, taking into account the increased presence of so called stegomalware [3, 4].

More precisely, this paper deals with the network of Cisco switches interconnected between themselves, that have enabled VTP protocol on them. In a nutshell, new covert channels deploy a combination of Switched spoofing VLAN attack (a kind of VLAN hopping) [5], and a version of "VTP bomb" attack

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), 13(3):37-49, Sept. 2022
DOI:*10.22667/JOWUA.2022.09.30.037*

*Corresponding author: Faculty of Computer Science, University "Goce Delcev", "Krste Misirkov" 10-A, Stip, 2000, Republic of N. Macedonia, Tel: +389-32-550-106

[6]. This is followed by transferring and storing secret data, that, for example, can be used in data exfiltration scenarios. The novel network steganografic methods affect the switches in the distribution and access layer of the three - tier hierarchical network model. Therefore, in this paper, we make three key contributions:

- We propose two novel network steganographic methods that store the secret message into all switches from the entire VTP domain. They create an indirect covert channels between the covert sender and covert receiver.
- We use the VTP protocol for the first time for hiding data.
- We perform a proof-of-concept implementation of the novel hiding method, followed by an experimental evaluation.

The rest of the paper is structured as follows. Sect. 2 introduces basics about VLANs, VTP and DTP functioning necessary to understand the new covert channels and discusses the related work. Sect. 3 describes the novel steganographic methods. Sect. 4 presents the experimental scenario, and gives some steganographic measures of the newly created covert channels, such as bandwidth and undetectibility. Countermeasures are given in Sect. 5, while the concluding remarks are in Sect. 6.

2 Basics and Related Work

2.1 VLANs

Virtual Local Area Networks (VLANs) are logical groups of end devices independent of their physical location, that act as a physical LANs. Each VLAN forms a separate broadcast domain, and can be spread across multiple switches, while on the same switch, several VLANs can exist and each VLAN can contain a subset of the switch ports (assigned on a port-by-port basis). This is possible because, each VLAN is associated with a separate IP subnetwork. So, traffic between different VLANs must be routed. By using VLANs, end devices can be grouped according to other criteria (e.g., department, type of users) instead of by physical location.

Cisco IOS Release 15.4SY supports 4096 VLANs in accordance with the IEEE 802.1Q standard [7]. The VLANS with IDs of 0 and 4095 are reserved only for the system use, while others are organized into two ranges:

- normal range - VLANs with IDs from 1 to 1005. VLANs with IDs 1, and 1002-1005 are reserved, can be used, but cannot be deleted.
- extended range - VLANs with IDs from 1006 to 4094.

Also a VLAN from normal range can be a private VLAN, and switch ports that belong to it are isolated from other switch ports, and can communicate only through a single uplink, usually connected to the router, server or other similar device.

Cisco IOS allows the maximal length of 32 characters for the VLAN's names, with a recommended length of 20 characters. For exceeding the recommended length, an informational message is generated.

VLANs can be manually created, deleted or modified on a given network device such as switch. In a large network, switches can automatically obtain and configure VLANs information, by using VLAN Trunking Protocol. The VLAN database is located in the flash memory in the form of a binary *vlan.dat* file. One important note here is that extended range VLANs are not written in the vlan.dat, but they are, by default, saved in the running configuration file.

2.2 VLAN Trunking Protocol (VTP)

VLAN Trunking Protocol (VTP) is a proprietary Cisco protocol [6, 8] that dynamically propagates VLANs information to all the switches in the switched network, reducing the administration of the network. It ensures that all switches within the same VTP domain have consistent VLAN configurations, and adding a new switch to the domain is simple, because it dynamically inherits VLAN information once connected. VTP is available on most of the Cisco switches, for example, switches from Catalyst series products. Currently, there are three versions of VTP - 1, 2 and 3.

VTP is a Layer 2 messaging protocol, and its scope is within a VTP domain. Each VTP domain consists of one or more network devices that share the same VTP domain name, and that are connected with Inter-Switch Link (ISL) and/or IEEE 802.1Q trunk interfaces (they carry traffic of all VLANs). When the administrator adds, deletes or modifies some VLAN on one VTP server, the new info is distributed through all switches in the VTP domain, as a VTP advertisement with increased configuration revision number.

Each switch can belong in at most one VTP domain at a given time. If the switch does not belong to any VTP domain, when it obtains the first VTP advertisement over a trunk link, it automatically becomes a member of the advertised VTP domain. Also, if the switch obtains several VTP advertisements from its VTP domain, it synchronizes with the VTP advertisement of the highest configuration revision number. The switch ignores VTP advertisements with a different VTP domain name or older (lower) configuration revision number.

All versions propagate configuration information for normal-range VLANs, while only VTP version 3 supports extended-range VLANs. Also, versions 1 and 2 do not propagate a private VLAN configuration, while version 3 propagates private VLAN configurations automatically.

VTP modes. The default VTP mode is server, and in this mode one can make VLAN's changes and can specify different VTP configuration parameters. If this results in the VTP advertisement with the highest configuration revision number, the changes will be honored by all the switches in its VTP domain. If the switch is configured in the transparent mode, the changes made on it, affect only that switch and are not advertised to others. It does not change its info with received VTP advertisements, but in VTP version 2, it forwards them to other switches. In the client VTP mode, the switch cannot make any changes on its VLANs, only it can synchronize with the VTP updates.

VTP version 3 differs between primary and secondary server modes. Only primary server, one per VTP domain, can update VLAN database information, and can send VTP advertisements to other switches. Secondary servers can not do that, but they can be promoted to the role of the primary server. Also, version 3 introduces the off mode, which is similar to the transparent mode, with one difference - the switch drops all the VTP messages and it does not forward them to other switches.

VTP messages. VTP messages are transported over the default VLAN, and over all trunk ports. The routers ignore the VTP messages. For demonstrating the idea of the paper, only versions 1 and 2 messages are explained. There are four types of VTP messages: summary advertisements, subset advertisements, advertisement requests and VTP join messages.

The switch in the server mode sends summary advertisements every 5 minutes to its neighboring switches (Figure 1,[9]). Switches in the same VTP domain (Management Domain Name field), compare the configuration revision number, and if the received one is higher than the one currently in possession, they will send advertisement requests for the most recent VLAN info. In other cases, the summary advertisement will be ignored. The Updater Identify field from the summary advertisement identifies the IP address of the management VLAN for the switch that made the last changes (or it gives 0.0.0.0 if the management VLAN is not set), while the Update Timestamp field is set to the time when this took place. If a password is configured for the VTP domain to prevent malicious switches to change the network, it must be set to all switches from that VTP domain. The 128-bit MD5 Digest field is calculated over the

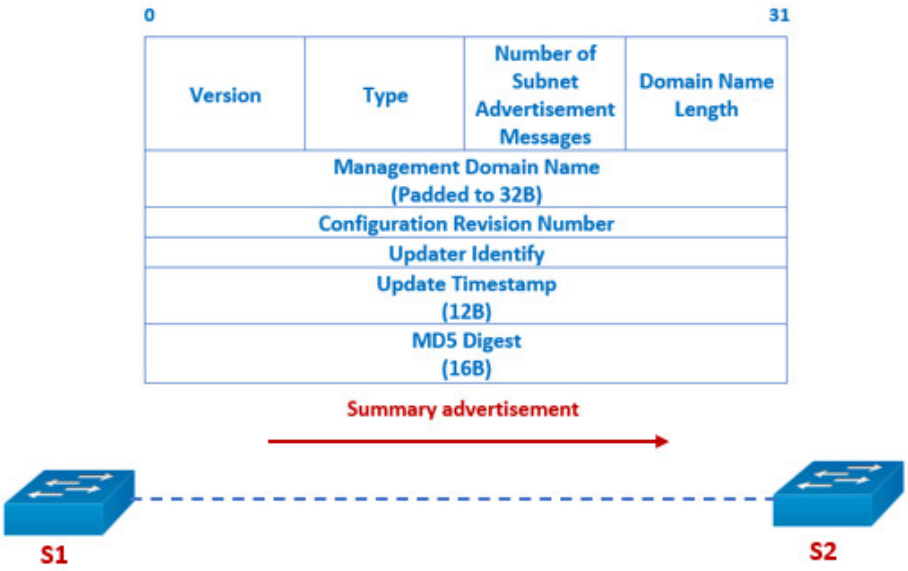

Figure 1: Switch S1 is sending summary advertisement message to its neighbour S2

VLAN database content and the VTP password, and it is transferred in all summary advertisements for validation of the updates. If the MD5 digest does not equal the one recalculated by the receiving switch, the received VTP messages would be dropped without any changes. In VTP version 1 and 2, passwords are stored in clear form on each switch, while in version 3, they are stored in encrypted form.

The advertisement request message is also sent by the switch in the cases when it is reset, or its VTP domain membership is changed. As an answer to the advertisement request from neighboring switches, one or more subset advertisement messages are sent to them (Figure 2,[9]). Together they carry all the content of the VLAN database.

The Seq-Number field identifies the subset instance, and it always starts with 1. If the list with VLAN info is quite long, it will be sent over several subset advertisement messages, and their number is specified in the originated summary advertisement.

VTP join messages are used when VTP pruning is enabled, but because VTP messages are not limited by pruning, its explanation is out of scope of this paper.

2.3 Dynamic Trunking Protocol (DTP)

Dynamic Trunking Protocol (DTP) is a proprietary Cisco protocol [10] that is used for negotiation of the trunk creation on a link between two switches, and the trunk encapsulation type (ISL or IEEE 802.1Q). It is enabled by default on most of the Cisco switches (it is not supported on the non-Cisco devices). Several modes on a given switch port can be configured as administrative mode: access, trunk, dynamic auto, dynamic desirable and nonegotiate. If the switch port mode of one of the two switches is set to trunk, and if the neighboring switch port mode is set to trunk, dynamic auto (default of new Cisco switch series) or dynamic desirable (default of some old Cisco switch series), the two neighboring ports will obtain the trunk operational mode after negotiation, and the trunk will be created.

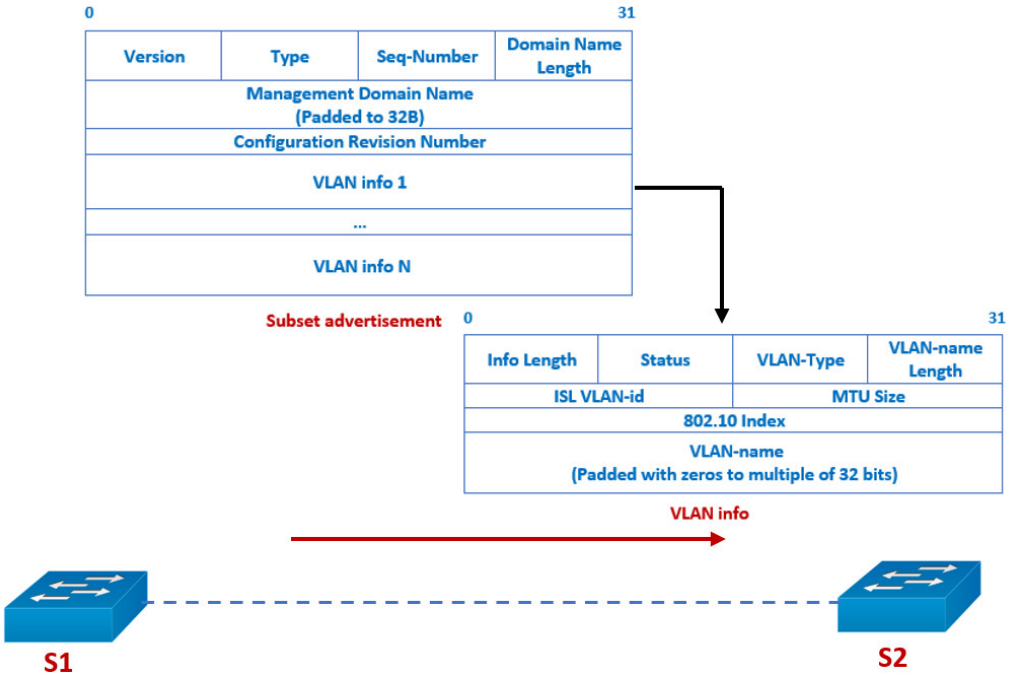

Figure 2: Switch S1 is sending subset advertisement message to its neighbour S2

2.4 Related Work

In this subsection, we are looking at three types of related work: VTP security issues, indirect network covert channels, and network covert channels that are using data-link or internet layer's protocols.

One of the biggest security issue with VTP deployment, especially for versions 1 and 2, is so called "VTP bomb", when an old switch is added to the VTP domain, with the higher configuration revision number, than revision numbers currently on the switches in the domain [6]. The old VLAN configuration will propagate to all the switches in the VTP domain, and any previous VLAN configuration on each switch will be wiped out and lost. VTP version 3 has a resolution for this issue, by allowing only switches with the role of primary server to update VLAN configuration of other devices in the VTP domain. The idea behind the "VTP bomb" is used for our new steganographic methods. The novelty of our approach lays in the fact that we use the switched network for hidden data storage and transfer, while the previous VLAN configuration is preserved and renewed, after the secret transfer finish.

VLAN hopping [5] is a type of attack that allows an attacker to bypass any layer 2 restrictions built to divide hosts connected in a switched network. Two basic forms of VLAN hopping are Switched spoofing VLAN attack and Double tagging (out of scope of this paper). Switched spoofing VLAN attack is possible when a malicious switch manages to create a trunk link with legitimate switch from a given switched network, and in the aftermath, it can eavesdrop on broadcast traffic on all VLANs on the network, or perform man-in-the-middle attacks [11]. We combine this attack with the VTP features, so, our switch can configure itself within a VTP domain, learning the current VLAN database for targeted network.

A good survey of indirect network covert channels can be found in [12], followed with proposed taxonomy. Here we will present several such covert channels, without an intention to be exhaustive.

One example of indirect covert channel is deploying an MQTT broker as an intermediate device for transferring hidden data between a publisher (as a covert sender) and one or more subscribers (as covert receivers), by publishing in a predefined ordered topics [13].

One can use the PACS archive as an intermediate node and DICOM C-MOVE service (for transferring medical images of any kind) to transfer a hidden data from one workstation or modality (as a covert sender) to other workstation (as a cover receiver) in or outside the hospital [14].

Several HTTP-based indirect covert channels can be found in literature (e.g., [15, 16]), where two HTTP servers exchange secret data through innocent clients. For that purpose, one server (as a covert sender) can force a client to redirect to other server (as a covert receiver), or can store a cookie in the client (later sent to the other server), etc.

Because there exist Layer 2 and Layer 3 switches, we are going to look at some covert channels deployed on data-link and internet layers. There are many surveys on network covert channels, such as [1, 2].

For shared medium networks (such as wireless LANs), Szczypiorski's HICCUPS system [17] hides data in the payload of the intentionally corrupted frames (with improper correction code values). Handel and Sandford [18] manipulate the CSMA/CD collision detection system, and by intentionally introducing or not collisions with frames from other users, they create one-bit per frame covert channel.

On the internet layer, several protocols can be used for hiding data. Most of the covert channels use some random, reserved or unused fields in the protocol header (e.g., Identification field of IPv4 [19], Target IP address of ARP [20], XID field of DHCP [21]), PDU payload (e.g., fragment payload in IPv4 [22], payload of ICMP Echo Request and ICMP Echo Reply packets [23]), packet or fragment sorting in IPv4 [24, 22], sending or not an IP packet in an arranged time interval [25], intentional IP packet losses [26], different rates for fragments [22], etc.

3 Proposed Methods

3.1 Communication Scenario

The communication scenario for the new method is very specific. There is no direct communication between the Covert Sender (CS) and the Covert Receiver (CR), but only through the network of interconnected switches (Figure 3). All interconnected switches in the same VTP domain are innocent intermediate nodes that store the secret data. One example where this can be deployed is in the organization with departments of different security clearances, and the same network switching infrastructure. So, CS can belong to the department with a higher security clearance and is subject to heavy security controls during going in/out of his office, while CR can work in the department with lower security clearance, not subjected to heavy security controls on the exit.

For CS and CR to be able to communicate secretly by this method, they need to know or learn the targeted VTP domain name, version and password. They also need to have access to the trunk, dynamic auto or dynamic desirable ports of some Cisco switch from the targeted network. We assume that at least switches from the distribution layer, and switches from access layer used by CS and CR are Cisco switches, with enabled VTP, and in the same VTP domain. Also, for the sake of simplicity, we will use only VTP version 1 or 2.

Both CS and CR have their switches, SCS and SCR. At the beginning, both switches SCS and SCR need to have a configuration revision number reset to zero, which can be done in several manners. For example, by changing the VTP mode in transparent, the configuration revision number will be reset to zero. After that the mode can be changed in client or server, according to CS' needs.

3.2 Description of the New Methods

Our new hiding methods are based on the fact that CS and CR can connect their switches with zero revision number and client mode to the legitimate switch of the specific VTP domain, which will result

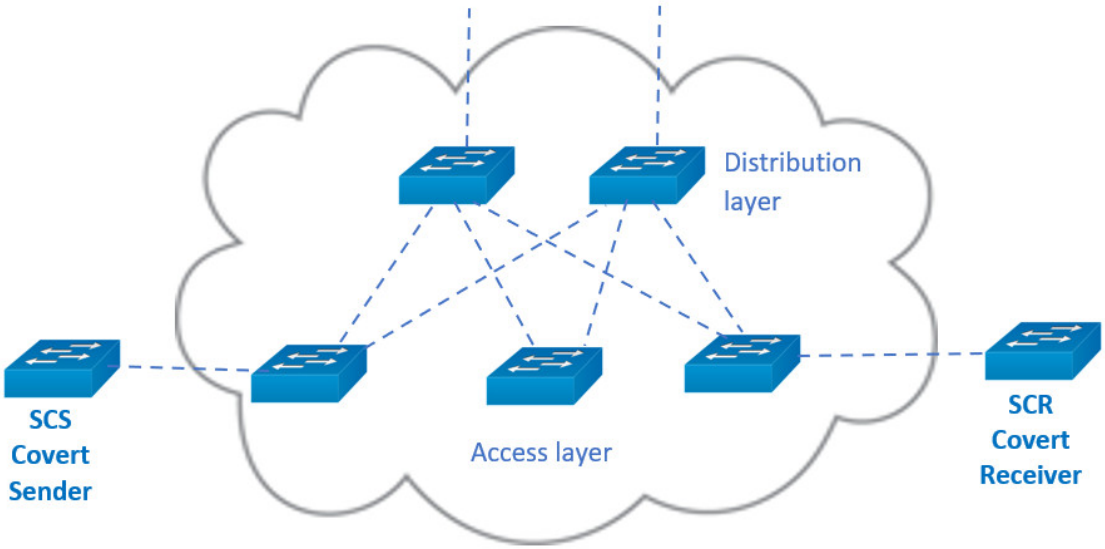

Figure 3: Communication scenario

in updating of the switches with the current VLAN configuration of the VTP domain. In order for this to be done, the first step is the creation of a trunk link between the stego switches and the legitimate switches. A switch port of SCS and SCR need to be configured as dynamic desirable or trunk, while the targeted ports of legitimate switches need to be configured by default dynamic auto, trunk or dynamic desirable. This means that DTP should be enabled on legitimate switches, and DTP negotiation will create the trunk. The second step is setting a proper VTP version, VTP domain and VTP password on the stego switches in the phase of learning. In this way, both CS and CR can learn the current VLAN configuration.

After CS has learned the current VLAN configuration and current configuration revision number (CRN), he/she disconnects the SCS from the network, changes its mode to server and makes changes in the VLAN configuration, for hiding the secret message. These changes increase the value of the CRN. After that, CS connects again the SCS to the switched network, and because it has the highest configuration revision number, its advertisements will be distributed across all switches, making them to become consistent with the SCS. In this way, the secret information is stored into all the switches from targeted network.

The CR connects its own SCR switch in some other part of the building as a client and with zero revision number, and the SCR is configured with the new configuration in at most 5 minutes. This is enough for CR to extract the secret message.

After that, CS can undo the performed changes on its SCS, and by connecting it again to the switched network, the new-old configuration will be propagated to all switches, because of the possession of highest CRN. The only difference from the start, will be the increased CRN. This entire process can be seen as one iteration. CS and CR can use more iterations for sending secret data.

We will work only with VLANs from normal range, with IDs from 2 to 1001, because only they are saved in vlan.dat. The CS can see which IDs from this range are not assigned. Usually, there are many IDs that are not assigned.

3.2.1 Embedding Method 1 (EM1) - Using VLAN Names

The secret message is encoded in the name of the newly created VLAN. Each name is at most 32 characters long, and the CS can create more VLANs like this, each carrying 32 characters of secret message. In this way, each newly created VLAN with name increases the CRN by 1.

3.2.2 Embedding Method 2 (EM2) - Creation/Deletion of New VLANs

The CS and CR decide which n free VLAN IDs are going to use, for example $V_1, V_2, \cdots, V_n$, given in their natural ordering by the value of the ID. The smallest ID from the chosen set is going to be V_1, while the largest ID is going to be V_n.

Let the CS wants to send a secret message M as a bit-array to the CR. He/she divides the message into the m_j blocks with length of n bits. The last block can be with the size smaller than n, so it can be padded with zeros till the full size.

The CS takes the switch SCS with learned VLAN configuration from the VTP domain, and a block $m_j = b_1 b_2 \cdots b_n$, and encode the secret message in the following way: if the $b_i = 1$, CS adds a VLAN with ID of V_i, else CS does not do anything. All the new VLANs can be created by one command, increasing the CNR only by 1.

3.3 Blind CS and CR

The CS and CR can learn the VTP version and VTP domain easily. For example, if CS puts the SCS in debug mode before connecting it to the legitimate switch, after the trunk is formed, this information is listed in the debug output. VTP password is not transferred in clear form, but together with vlan.dat it is hashed with MD5 hash function. Regardless that MD5 is a weak hash function with many known attacks, there is no way for a password to be deduced from it.

On the other hand, if vlan.dat is transferred somewhere in the network by TFTP for example, by capturing the network traffic, one can easily find the password from it. Even though vlan.dat is a binary file, if you open it in the Notepad++ for example, VTP domain and password are readable.

The other possibility is the VTP password to be learned from the network administrator.

3.4 Categorization

If we apply the new taxonomy for indirect network covert channels, suggested in [12], both methods can be classified as hybrid methods between the Redirector and Dead Drop patterns. This is the case because on one side we have legitimate switches as innocent intermediate nodes redirecting the secret message to the CR, while on the other side, the secret message is stored in all the switches from the targeted network.

4 Experimental Evaluation

4.1 Experimental Scenario

In order to have a proof-of-concept and to test our methods, we implemented a small network of two distribution layer switches DSW1 and DSW2, and two access layer switches ASW1 and ASW2 (Figure 4). DSW1 and DSW2 are VTP servers, and STP primary and secondary root switches. We used Cisco Catalyst 3650 Series L2/L3 switches with IOS XE 16.3.2 for DSW1 and DSW2. ASW1 and ASW2 are VTP clients, and we used their ports Fa0/1 set in dynamic auto mode, for connection with stego switches. For the test we used Cisco Catalyst 2960L Series L2 switch with IOS 15.2(7)E3 as ASW1 and ASW2.

VTP version 2 is running on all the switches. They are connected in the VTP domain *VTPTEST* with password *UGDStip*.

CS and CR are using the switches SCS and SCR, which are Cisco Catalyst 2960-X Series with IOS 15.2(2)E6. The Fa0/1 ports of both switches were set to trunk mode.

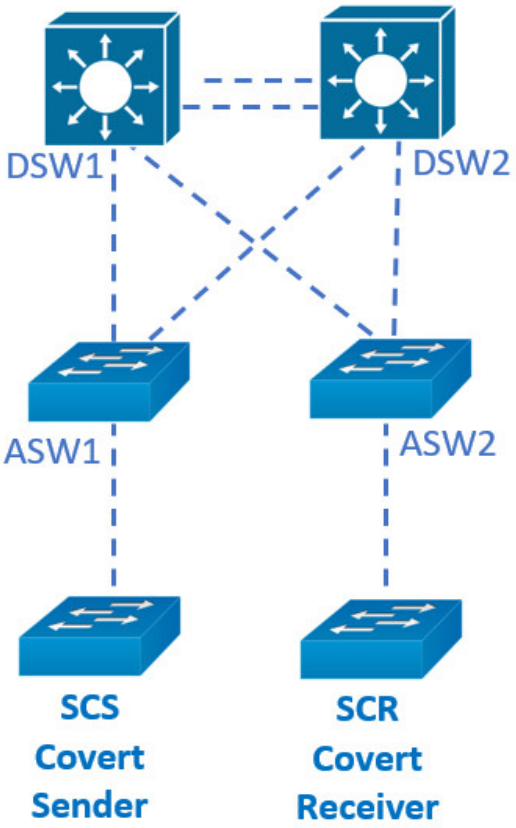

Figure 4: Experimental testbed

Legitimate network has three more VLANs beside the five reserved ones (1, 1002-1005), with names VTPTEST33, VTPTEST34 and VTPTEST35, and VLAN ID 33, 34 and 35, respectfully. The CRN in the moment of using was 446. Only 4 (1, 33, 34, 35) out of 8 existing VLANs participate in STP and there are only 4 STP instances.

First, we managed to connect SCS and SCR (with CNR reset to zero and VTP client mode) to appropriate Fa0/1 ports from ASW1 and ASW2, respectfully, and to negotiate trunk link. After setting the VTP version 2, and appropriate VTP domain and password, we managed to retrieve the VLAN configuration from our testbed.

After that, the SCS was disconnected from the network, to configure it with hidden data. We decided to use VLANs with IDs from 101 till 160, and in the worst case scenario, we wanted to transmit an array of 60 binary ones, which means all 60 VLANs are to be created, according to an EM2 method. Creation was performed with one command on the SCS, and CNR obtained a new value of 447. After that, SCS was configured as a VTP server and connected to ASW1 again. Now, because its CRN was the highest one, all the switches in the testbed obtained the new configuration, together with the SCR. CR managed to retrieve the hidden message.

Next we deleted newly created VLANs from the SCS with one command, increasing the CNR to

448. The new change also propagated to all the switches, resetting the old VLAN configuration, just with CNR of 448 instead of 446.

For probing the EM1 method, we repeated the procedure, by starting with CNR of 448, and created a VLAN with name *TESTTESTTESTTESTTESTTESTTESTTEST* to send the same message. At the end, when this VLAN was deleted, the original VLAN configuration was successfully restored, just with CNR value of 450.

4.2 Bandwidth and Undetectibility

Different Cisco switches have a different maximal number of VLANs that can be configured at the same time (that can exist in the VLAN database). Another limiting factor is the presence of Spanning Tree Protocol (STP) and its different versions [8]. In Per-VLAN Spanning Tree Plus (PVST+) or Rapid PVST+, each VLAN has its own STP instance, and there is a maximal number of STP instances that can be created. This number also differs for different Cisco switches. For example, Catalyst 2950 LRE switch has the maximum of 250 supported VLANs at the same time, while the maximum number of STP instances is 64 [1]. For a Catalyst 3650 switches the maximum number of STP instances is 128, while the maximal number of supported VLANs is 1005 [2]. So, if Multiple STP is not used, the main limiting factor for the number of supported VLANs is the maximum number of STP instances. If you exceed this number, an error message will be generated, similar to "SPANTREE_VLAN_SW-2-MAX_INSTANCE: Platform limit of 64 STP instances exceeded. No instance created for VLANxxx".

Because CS and CR want to stay low on the radar, and they do not want to rise any error messages, it is safe to go with at most 64 created STP instances (and VLANs) at the same time. This number includes default VLAN 1 and existing VLANs in the organizational network. Therefore, there are already K existing STP instances. The CS can create at most 64-K additional VLANs, which means that the bandwidth for EM1 method is at most 32(64-K) characters per one iteration, while for EM2 the bandwidth is 64-K bits per one iteration. One can choose to use only 20 characters per VLAN (so informational message will not show on the switch), lowering the bandwidth for EM1 to at most 20(64-K) characters per one iteration.

In our experimental testbed, because there are only 4 STP instances in the legitimate network, we have an opportunity to create up to 60 more VLANs (and STP instances) before hitting the limit of 64 STP instances. This means that the bandwidth of one iteration for our testbed is 32*60=1920 characters (or 20*60=1200 characters without informational messages) for EM1, and 60 bits for EM2.

The main problem with undetectibility of our new methods is the increased value of CRN, which cannot be undone, because it is 32 bits long, and increases by one per each change of VLANs structure. The good thing is that the creation or deletion of several VLANs (without naming them) can be done with one command, which increases the CRN only by 1. On the other hand, if you create several VLANs with names, each newly created VLAN will increase the CRN by 1.

The best way to deploy the EM1 method, is to increase the current value of CRN, C, only by 1. For this to happen, CS needs to have 2 switches, one for storing the legitimate VLAN configuration, and other to create VLANs, that will contain all legitimate VLANs, and newly created VLANs till the C+1 is reached. After the transfer to CR is performed, CS will use the first switch and will create and delete one VLAN. In this way the CRN will become C+2, and original configuration will be restored.

Another trace of the performed changes on the switches can be found in the status of VTP. The status shows when was the configuration last modified and from which IP address.

If the Simple Network Management Protocol (SNMP) management is not configured to catch events like VLAN creation or deletion, the created changes would not rise any alerts.

[1] Catalyst 2950 Desktop Switch Software Configuration Guide, 12.1(11)YJ4, 2019

[2] Layer 2/3 Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3650 Switches)

5 Countermeasures

The basic way to prevent any VTP-based covert channel, is VTP to be disabled in the network perimeter, the decision upon which many security experts mutually agree. Other basic countermeasure is DTP to be disabled on all legitimate switches.

If VTP must be enabled, then it is better to use VTP version 3, which allows only primary server to update VLAN database information and to send VTP advertisements to other switches. Still, the secondary servers can be promoted to the role of the primary server.

The SNMP traps can also be used to catch the events that happen on a switch during clandestine communication. For example, there are two Objects Identifiers: vtpVlanCreated (1.3.6.1.4.1.9.9.46.2.0.10) and vtpVlanDeleted (1.3.6.1.4.1.9.9.46.2.0.11) that can be used to send alert to the network administrators every time a VLAN is created or deleted.

Network administrators can track the CRN from time to time, and look for suspicious big values.

If there are routers in the distribution layer, instead of switches, VTP messages would be ignored and not farther distributed, by which the VTP-based covert channel will be limited only one part of the access layer.

It is clear, that VTP-based covert channel can not exist in the network of non-Cisco switches, which do not support VTP at all. So, by using non-Cisco switches in the distribution layer can limit the covert channel only one part of the access layer.

6 Conclusion

In this paper, we present two novel steganographic methods that use an entire network of Cisco switches as innocent intermediate nodes, for creation of indirect covert channels between CS and CR. These methods will work even in the case of a presence of non-Cisco switches, though there is a requirement that the switches from distribution layer and both switches from access layer are to be used from the CS and CR to be Cisco switches. Another requirement is VTP to be enabled on these switches. If the connection ports from legitimate switches are not in trunk mode, then DTP should also be enabled on these switches (only the access mode is unwanted).

While there are many known storage and timing network covert channels suggested for data-link and internet layer protocols, this is the first time that VTP is used as a carrier.

For future reference, we are planning to see how VTP version 3 influences our steganographic methods. Also, there is another protocol Multiple VLAN Registration Protocol (MVRP), which is part of the Multiple Registration Protocol (MRP). MVRP is standardized in IEEE 802.1ak amendment to the IEEE 802.1Q standard. So, it will be interesting to see how this protocol can be used for creating covert channels.

References

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

[2] A. Mileva and B. Panajotov. Covert channels in the tcp/ip protocol stack – extended version. *Open Computer Science*, 4(2):45–66, June 2014.

[3] G. Suarez-Tangil, J. E. Tapiador, and P. Peris-Lopez. Stegomalware: Playing hide and seek with malicious components in smartphone apps. In *Proc. of the 10th International Conference of Information Security and Cryptology (Inscrypt'14), Beijing, China*, volume 8957 of *Lecture Notes in Computer Science*, pages 496—-515. Springer, December 2014.

[4] K. Cabaj, L. Caviglione, W. Mazurczyk, S. Wendzel, A. Woodward, and S. Zander. The new threats of information hiding: the road ahead. *IEEE IT Professional*, 20(3):31–39, June 2018.

[5] AT&T Cybersecurity blog PAM. Vlan hopping: How to mitigate an attack, December 2019. `https://cybersecurity.att.com/blogs/security-essentials/vlan-hopping-and-mitigation` [Online; Accessed on September 15, 2022].

[6] Cisco. Supervisor engine 2t software configuration guide, release 15.4sy, chapter 25. vlan trunking protocol(vtp), August 2019. `https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst6500/ios/15-4SY/config_guide/sup6T/15_3_sy_swcg_6T/vtp.pdf` [Online; Accessed on September 15, 2022].

[7] IEEE 802.1 Working Group. 802.1q-2014, bridges and bridged networks, December 2014. `https://www.ieee802.org/1/pages/802.1Q-2014.html` [Online; Accessed on September 15, 2022.

[8] N. Kocharians and P. Paluch. *CCIE Routing and Switching v5.0 Official Cert Guide*. Cisco Press, 2014.

[9] Louis D. Rossi. *Cisco Catalyst LAN Switching*. McGraw-Hill, 1999.

[10] Cisco Networking Academy. Routing and switching essentials companion guide. dynamic trunking protocol(3.2.3), April 2014. `https://www.ciscopress.com/articles/article.asp?p=2181837&seqNum=8` [Online; Accessed on September 15, 2022].

[11] I. Pepelnjak. Omg: Vtp is insecure, February 2022. `https://blog.ipspace.net/2022/02/vtp-insecure.html` [Online; Accessed on September 15, 2022].

[12] T. Schmidbauer and S. Wendzel. Sok: A survey of indirect network-level covert channels. In *Proc. of the 17th ACM ASIA Conference on Computer and Communications Security (Asia CCS'22), Nagasaki, Japan*, pages 546—-560. ACM, May 2022.

[13] A. Velinov, A. Mileva, S. Wendzel, and W. Mazurczyk. Covert channels in the mqtt-based internet of things. *IEEE Access*, 7:161899—-161915, November 2019.

[14] A. Mileva, A. Velinov, V. Dimitrova, L. Caviglione, and S. Wendzel. Information hiding in the dicom message service and upper layer service with entropy-based detection. *Entropy*, 24(2):176, January 2022.

[15] M. Bauer. New covert channels in http: adding unwitting web browsers to anonymity sets. In *Proc. of the 2003 ACM Workshop on Privacy in the Electronic Society (WPES'03), Washington, DC, USA*, pages 72–78. ACM, October 2003.

[16] G. Daneault and D. Johnson. Client-initiated http covert channels using relays. In *Proc. of the 4th International Symposium on Digital Forensic and Security(ISDFS'16), Little Rock, AR, USA*, pages 32–37. IEEE, April 2016.

[17] K. Szczypiorski. Hiccups: Hidden communication system for corrupted networks. In *Proc. of the 10th International Multi-Conference on Advanced Computer Systems (ACS'2003), Miedzyzdroje, Poland*, pages 31–40. Citeseer, January 2003.

[18] T. G. Handel and M. T. Sandford. Hiding data in the osi network model. In *Proc. of the 1st International Workshop on Information Hiding, Cambridge, United Kingdom*, Lecture Notes in Computer Science, pages 23–38. Springer-Verlag, June 2005.

[19] C. H. Rowland. Covert channels in the tcp/ip protocol suite. *First monday*, 2(5), May 1997.

[20] L. Ji, Y. Fan, and C. Ma. Covert channel for local area network. In *Proc. of the 2010 IEEE International Conference on Wireless Communications, Networking and Information Security (WCNIS'10), Beijing, China*, pages 316–319. IEEE, June 2010.

[21] R. Rios, J. A. Onieva, and J. Lopez. Hide dhcp: Covert communications through network configuration messages. In *Proc. of the 27th IFIP TC 11 Information Security and Privacy Conference (SEC'12), Heraklion, Crete, Greece*, volume 376 of *IFIP Advances in Information and Communication Technology*, pages 162–173. Springer, June 2012.

[22] W. Mazurczyk and K. Szczypiorski. Steganography in handling oversized ip packets. In *Proc. of the 2009 International Conference on Multimedia Information Networking and Security (MINES'09), Wuhan, Hubei, China*, pages 559–564. IEEE, November 2009.

[23] daemon9, AKA, and route. Project loki. *Phrack Magazine*, 49(article 6), 1996.

[24] R. Poudel. Covert channel analysis and data hiding in tcp/ip. ResearchGate, DOI:

10.13140/RG.2.2.21348.94087, January 2002. http://dx.doi.org/10.13140/RG.2.2.21348.94087 [Online; Accesed on September 15, 2022].

[25] M. A. Padlipsky, D. W. Snow, and P. A. Karger. Limitations of end-to-end encryption in secure computer networks, August 1978. https://readings.owlfolio.org/1978/limitations-end-to-end-encryption/ [Online; Accessed on September 15, 2022].

[26] S. D. Servetto and M. Vetterli. Communication using phantoms: covert channels in the internet. In *Proc. of the 2001 IEEE International Symposium on Information Theory (ISIT'01), Washington, DC, USA*, page 229. IEEE, June 2001.

Author Biography

Aleksandra Mileva received her Master and PhD degrees from Ss. Cyril and Methodius University, Skopje, N. Macedonia, in 2001 and 2010, respectively. Currently, she is a professor at the Faculty of Computer Science, University "Goce Delcev," Stip, N. Macedonia, and the Head of the university's Laboratory of Computer Security and Computer Forensics. Her research interests include computer and network security, digital steganography, IoT security, cryptography, computer forensics, and quasigroups theory. Since 2019, she has been a member of the EURASIP Data Forensics and Security TAC. She was with the management committee of two COST actions IC1201: BETTY and IC1306: Cryptography for Secure Digital Interaction, and she was a member of the Advisory board of H2020 SIMARGL project. She is a co-author and developer of the NaSHA family of hash functions, which where the First Round Candidate of the NIST SHA-3 Competition (2007-2012).

Jordan Tikvesanski is Deputy Head of ICT at University "Goce Delcev," Stip. He received his BSc degree in Computer Science at University "Goce Delcev" and he holds CCNA and CCNP Enterprise certificates. With over 20 years of experience in network and system engineering, and leading the process of designing, planning, and implementing the LAN and WAN network of the University "Goce Delcev," he was engaged in various national and international projects as a network and system engineer providing expertise in designing complex highly available datacenter solutions, wired and wireless networks, VPNs, both on-premises and in the cloud.