TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies

Hindrek Baum 163417IVCM

DETECTION OF VLAN HOPPING ATTACKS USING SWITCH'S MONITORING OPTIONS

Master’s Thesis

Supervisor: Jaan Priisalu
MSc

Tallinn 2018

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Hindrek Baum 163417IVCM

KOHTVÕRGU VIRTUAALSEGMENDI HÜPITUSRÜNNETE TUVASTAMINE RAKENDADES KOMMUTAATORI VÕIMALUSI

Magistritöö

Juhendaja: Jaan Priisalu
MSc

Tallinn 2018

Author’s declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Hindrek Baum

07.05.2018

Abstract

Ethernet switches are most accessible devices in the organization, and various malicious insiders can easily get access to Ethernet switches. Ethernet switches are vulnerable to various attacks including VLAN hopping attacks.

VLAN hopping attacks are hard to discover because the attacker is exploiting the switch internally. There is information on how to execute VLAN hopping attacks and how to harden the switch against VLAN hopping attacks. Almost no information is available on the detection side of the VLAN hopping attacks.

The objective of the thesis is to analyse detection possibilities of the VLAN hopping attacks at the Ethernet switch level by using a Syslog, SNMP and Port Mirroring options of the switch.

This thesis is written in English and is 93 pages long, including 9 chapters, 25 figures and 20 tables.

Annotatsioon

Kohtvõrgu virtuaalsegmendi hüpitusrünnete tuvastamine rakendades kommutaatori võimalusi

Kohtvõrgu kommutaatorid on kergesti avatud sisemistele rünnetele. Virtuaalsete kohtvõrkude eesmärk on parandada turvalisust kommutaatori tasemel, kuid virtuaalsegment ise on avatud mitmetele rünnetele.

Töö käigus uuritakse kohtvõrgu virtuaalsegmendi vastu suunatud hüpitusrünnete avastamise võimalusi. Kuna hüpitusrünne toimub kommutaatori sees, siis on seda väga raske avastada. Töö käigus püütakse uurida, kas ründe toimumise hetkel on üldse on võimalik tuvastada, et rünne toimub või on juba toimunud ning millisel moel jätavad erinevad virtuaalsegmendi hüpitusründed endast jälgi kommutaatori infokanalitesse?

Virtuaalsete kohtvõrkude vastu suunatud hüpitusrünnete avastamise võimalusi ei ole piisavalt uuritud. On infot selle kohta, kuidas ründeid teha ja kuidas konfigureerida kommutaatoreid turvaliseks, kuid ei leidu piisavalt infot selliste rünnete tuvastamise võimaluste kohta.

Uuritakse kommutaatori tasemel seadistatavate süsteemi logide, lihtsa võrguhalduse protokolli ja peegelpordi kasutamise võimalusi tuvastada kommutaatori vastu suunatud hüpitusründeid.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 93 leheküljel, 9 peatükki, 25 joonist, 20 tabelit.

List of abbreviations and terms

ACL	Access Control Lists
ARP	Address Resolution Protocol
CAM	Content Addressable Memory
CDP	Cisco Discovery Protocol
CIDR	Classless Inter-Domain Routing
CLI	Command Line Interface
CME	Cisco Unified Communications Manager Express
CNA	Cisco Network Assistant
CVE	Common Vulnerabilities and Exposures
DAI	Dynamic ARP Inspection
DHCP	Dynamic Host Configuration Protocol
DTP	Dynamic Trunking Protocol
ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IOS	Internetwork Operating System
IPSG	IP Source Guard
ISL	Inter-Switch Link
ISO	International Organization for Standardization
KVM	Kernel-based Virtual Machine
LAN	Local Area Network
MAC	Media Access Control
MIB	Management Information Base
NIC	Network Interface Card
NMS	Network Management Stations
NVRAM	Non-Volatile Random-Access Memory
OID	Object Identifier
OS	Operating System

OSI	Open Systems Interconnection
PAgP	Port Aggregation Protocol
PCAP	Packet Capture
PVLAN	Private Virtual Local Area Network
QoS	Quality of Service
RFC	Request for Comments
RTP	Real-time Transport Protocol
SCCP	Skinny Client Control Protocol
SIP	Session Initiation Protocol
SNMP	Simple Network Management Protocol
SPAN	Switched Port Analyser
SVI	Switch Virtual Interface
TCAM	Content-Addressable Memory
TCP	Transmission Control Protocol
TUT	Tallinn University of Technology
UDP	User Datagram Protocol
UTP	Unshielded Twisted-Pair
VLAN	Virtual Local Area Network
VMPS	VLAN Management Policy Server
VoIP	Voice over IP
VTP	VLAN Trunking Protocol

Table of contents

1 Introduction 12
1.1 Problem statement 12
1.2 Objective of the thesis 14
1.3 Research method 16
1.4 Scope and limitations 17
1.5 Acknowledgements 17
2 Data Link layer 18
2.1 A Layer 2 Ethernet switch operation 20
2.2 VLANs and 802.1Q trunking 22
2.3 Private VLANs 24
2.4 Voice VLAN 26
3 Switch monitoring options 28
3.1 Syslog 28
3.2 Simple Network Management Protocol 30
3.3 Packet captures 32
4 VLAN hopping attacks 34
4.1 VLAN security issues 35
4.2 Switch Spoofing attack / basic VLAN hopping attack 37
4.3 Double Encapsulated 802.1Q VLAN hopping attack 38
4.4 Private VLAN (PVLAN) hopping attack 40
4.5 Voice VLAN (VoIP) hopping attack 41
5 Research model 43
6 Applying the research model to VLAN hopping study 46
6.1 Preparation 47
6.2 Detection phase 49
6.3 Collection phase 50
6.4 Analysis and investigation phases 51
6.5 Presentation phase 52
7 Testing environment 54

7.1 Testing objectives 54
7.2 Network topology 56
7.2.1 Addressing table 58
7.3 Attack tools 59
7.3.1 Yersinia 59
7.3.2 PackETH 60
7.3.3 VoIP Hopper 61
7.4 Testing procedure 61
8 Attacks and results 63
8.1 Switch Spoofing attack / Basic VLAN hopping attack 63
8.1.1 SNMP and Syslog results 64
8.1.2 Traffic attributes 64
8.1.3 Compare packet capture methods 65
8.2 Double Encapsulated 802.1Q VLAN hopping attack 67
8.2.1 SNMP and Syslog results 68
8.2.2 Traffic attributes 68
8.2.3 Compare packet capture methods 69
8.3 Private VLAN (PVLAN) hopping attacks 70
8.3.1 Inside isolated PVLAN attacks 70
8.3.2 Attacks between the community and isolated PVLANs 72
8.3.3 SNMP and Syslog results 72
8.3.4 Compare packet capture methods 72
8.4 Voice VLAN (VoIP) hopping attack 73
8.4.1 SNMP and Syslog results 74
8.4.2 Compare packet capture methods 74
9 Summary 76
References 83
10 Appendix 1 – Switch configurations 87
11 Appendix 2 – Router configurations 90
12 Appendix 3 – SPAN configurations 92
13 Appendix 4 – Debugging and troubleshooting commands 93

List of figures

Figure 1: OSI model compared to the two TCP/IP models. 19
Figure 2: The basic IEEE 802.3 MAC data frame format. 19
Figure 3: Management, Control and Data planes. 20
Figure 4: Layer 2 switch operation mechanics. 21
Figure 5: IEEE 802.1Q trunking. 22
Figure 6: Private VLAN domain and secondary VLANs 25
Figure 7: Private VLAN traffic flows 25
Figure 8: Separating voice and data traffic using VLANs. 27
Figure 9: Syslog message severity levels by keyword and numeral. 29
Figure 10: SNMP version 1 overview. 31
Figure 11: Basic VLAN hopping attack. 37
Figure 12: Ethernet frame with two IEEE 802.1Q tags. 39
Figure 13: Double 802.1Q encapsulation VLAN hopping attack. 39
Figure 14: Private VLAN attack description. 40
Figure 15: VoIP implementation 41
Figure 16: A Generic Process Model for network forensics. 44
Figure 17: Proposed model change. 47
Figure 18: Applied Generic Process Model for network forensics. 52
Figure 19: Network topology. 57
Figure 20: Establishing the trunk link. 63
Figure 21: An example of captured DTP frame. 65
Figure 22: Editing 802.1Q fields in Yersinia. 68
Figure 23: An example of the double-encapsulated frame on the attacker PC. 69
Figure 24: The double-encapsulated frame after the first VLAN tag is stripped off. 69
Figure 25: The double-encapsulated frame at the final destination. 69

List of tables

Table 1: Brief overview of CVE entries. 35
Table 2: The combinations of DTP modes. 38
Table 3: Used hardware and software. 56
Table 4: Addressing table 58
Table 5: The Switch Spoofing attack success states 63
Table 6: An SNMP and Syslog reactions to Switch Spoofing attacks 64
Table 7: SPAN and VSPAN results of the Switch Spoofing attacks. 65
Table 8: VLAN filtering results of the Switch Spoofing attacks. 66
Table 9: SPAN results of the Double Encapsulated VLAN hopping attacks. 67
Table 10: VLAN filtering results of the Double Encapsulated VLAN hopping attacks. 70
Table 11: The isolated PVLAN testing topology. 71
Table 12: An example of the modified malicious frame. 71
Table 13: The community and isolated PVLAN testing topology. 72
Table 14: SPAN and VSPAN results of the PVLAN hopping attacks. 72
Table 15: Voice VLAN testing topology. 74
Table 16: SPAN and VSPAN results of the Voice VLAN hopping attacks. 75
Table 17: VLAN filtering results of Voice VLAN hopping attacks. 75
Table 18: A comparison of different VLAN hopping attacks. 78
Table 19: An SNMP and Syslog reactions to VLAN hopping attacks. 79
Table 20: A comparison of different SPAN options to detect VLAN hopping attacks. 80

1 Introduction

The thesis will give an overview of the various virtual local area network (VLAN) attacks. The research tries to find out how the Ethernet switch is reacting to these attacks and whether these attacks are detectable by using various monitoring options of the switch.

1.1 Problem statement

Ethernet switches are common network devices which are easily accessed from inside the organization. External attackers can also use social engineering to get access to Ethernet switches, but many insiders are onsite malicious employees. After the Ethernet switch is compromised, it is much easier to build attacks against upper layer protocols [1].

Insider Threat Report 2018 surveyed 472 cybersecurity professionals which revealed that 90% of organizations think that they are vulnerable to insider attacks and 53% of the respondents confirmed insider attacks against their organisation in the previous 12 months [2, p. 4].

Ethernet switches are operating primarily on the Data Link layer of the OSI model. The design of modern network devices is referencing the Open Systems Interconnection (OSI) networking model. The OSI networking model is designed and built in a way that different layers can work independently without the knowledge of other layers which makes targeted attacks against Data Link Layer invisible to upper or lower layers. If one layer is compromised, then other layers may not be aware of the situation [3, p. 277].

As mentioned in [4, p. 81]: “Security is only as good as the weakest link and if the weakness is at a low level in the communication stack then every other Layer has potential to inherit the problem.”

As shown by Altunbasak et al. in [5], the Data Link layer of the OSI model can be considered as the weakest link in network security. Another study [6] described the

security weaknesses of the OSI model by presenting the weak links between Data Link layer and Network layer of the OSI model.

This context illustrates the insider attack problems which are affecting the internal network security of the organisations. When the Ethernet switches are left to their default configurations, the situation becomes more vulnerable. If the switches are set up with mainly their factory default settings, they will be leaving a lot of well-known vulnerabilities wide open to attackers [7, p. 4].

As stated in [8, p. 156]: "A Cisco Catalyst switch comes from the factory ready to switch frames. All you have to do is connect the power cable, plug in the Ethernet cables, and the switch starts switching incoming frames. Connect multiple switches together, and they are ready to forward frames between the switches as well. Moreover, the big reason behind this default behaviour has to do with the default settings on the switches. Cisco Catalyst switches come ready to get busy switching frames because of these default settings:

- The interfaces are enabled by default, ready to start working once a cable is connected.
- All interfaces are assigned to default VLAN 1.
- 10/100 and 10/100/1000 interfaces use "autonegotiation" configuration parameter by default, which enables to automatically establish trunk links.
- The MAC learning, forwarding, flooding logic all works by default.
- Spanning Tree Protocol is enabled by default."

A Local Area Network (LAN) includes all devices in the same broadcast domain, which includes the set of all LAN-connected devices. When any of the devices in LAN sends a broadcast frame, then all the other devices get a copy of the frame. Virtual Local Area Network (VLAN) technology enables to segment the single physical switch into multiple separated virtual broadcast domains. If there are no VLANs, then all switch interfaces are in the same broadcast domain. VLANs provide logical segmentation where a single switch can have some interfaces in one virtual broadcast domain and some in another virtual broadcast domain [8, pp. 244–5].

As stated in [9], the logically separated and isolated VLANs should provide better security, but various research papers revealed that VLANs are vulnerable to many

attacks known as VLAN hopping attacks [3], [10], [11], [12], [13]. VLAN hopping enables to gain unauthorised access to another VLAN at Ethernet switch level without using OSI Network layer device which is usually a router. VLAN hopping causes a situation where ingress and egress interfaces of the switch belong to different VLANs. VLANs are configured at the Ethernet switch level and VLAN hopping is happening inside the switch. Therefore VLAN hopping attacks are particularly hard to detect.

There are a large variety of vendors who are providing Ethernet switches. Cisco has over 50% of a global market share of Ethernet switches [14].

Although the switches can be hardened to provide more security, there are still zero-day vulnerabilities, and security administration involves administrative burden.

1.2 Objective of the thesis

Multiple articles focus on the Data Link layer security issues and also cover VLAN hopping attacks.

Most of the work is done on the attack and defence side including various recommendations for switch hardening. However, there is almost no knowledge of the detection side of VLAN hopping attacks.

The VLAN hopping attacks are hard to discover because the attacker is exploiting the switch internally. Because VLAN hopping is happening inside the switch, then one possible way for the detection is to use switch's monitoring sources. Typical Cisco switch has three built-in data sources for the monitoring:

- Syslog - Cisco IOS software is supporting the error and system messages [15].
- SNMP (Simple Network Management Protocol) - there are various traps available that are related to VLANs, MAC address table and other features of the switch [16].
- SPAN (Switched Port Analyser) - the SPAN feature is also named as a port monitoring or port mirroring which is used to acquire network traffic for analysis by a network analyser. It enables to monitor Ethernet port(s) or group of different VLANs [17].

The switch can be configured to act as an SNMP or Syslog agent to send internal status information to the external server which helps to identify the possible malicious events.

The port mirroring enables to capture network traffic from the switch which might contain attributes to identify the malicious activities. The port mirroring itself has four different options for the monitoring:

- Physical Ports;
- Trunk Ports;
- VLAN based local SPAN (VSPAN) which enables to monitor all active ports of the trunk;
- VLAN filtering of the trunk ports which enables to monitor allowed VLANs of the trunk.

The local SPAN has various constraints for the data acquisition. By default, the local SPAN is not mirroring OSI layer 2 traffic and not preserve 802.1Q encapsulations. Appropriate SPAN configuration parameters are required to mirror OSI layer 2 protocols [17].

According to the Cisco [18], the SPAN traffic is handled with lower priority compared to the regular traffic of the switch. The switch might drop the Ethernet packets due to congestions or oversubscribed destination ports.

Mirroring each physical port has a scalability issue because it is difficult to mirror many switch ports or to deploy many sniffers into the network.

The objective of the thesis is to research how the various monitoring options of the switch can be used for the detection of VLAN hopping attacks?

The hypothesis of the thesis specifies that different types of VLAN hopping attacks which are exploited internally of the switch left behind artefacts, patterns or attributes which can be detected by using available monitoring options of the switch.

The first research objective of the thesis questions whether the VLAN specific VSPAN or VLAN filtering options are more suitable and scalable choices for the monitoring? Because the particular VLANs or trunk links are mirrored directly, the VSPAN and VLAN filtering options should perform better to detect various VLAN hopping attacks.

Mirroring only the physical ports enables to receive the ingress traffic, but VSPAN or VLAN filtering allows the monitoring at the VLAN level. Because multiple VLANs can be monitored at the same time, these VSPAN or VLAN filtering methods should be more suitable to observe VLAN hopping attacks.

The second objective of the thesis is to identify how the Syslog and SNMP are reacting to the different VLAN hopping events. What kind of SNMP trap messages and Syslog messages are displayed during VLAN hopping events?

The research tries to find out what is happening inside the switch at the time of different VLAN hopping events.

1.3 Research method

The problem statement and research questions are indicating that for investigating detection possibilities by analysing reactions, traces, patterns of different VLAN hopping attacks, it is necessary to simulate these attacks in a laboratory environment.

The research requires repetitive actions to execute multiple VLAN hopping attacks to test various port mirroring configurations, therefore it is essential to have a research model to determine the whole research process.

These objectives and activities are inherently referring to the network forensic analysis. As highlighted in [19, p. 15]: "The concept of network forensics deals with data found across a network connection mostly ingress and egress traffic from one host to another. Network forensics is the science that deals with capture, recording, and analysis of network traffic. The network log data are collected from existing network security products, analysed for attack characterisation and investigated to trace back the perpetrators."

Pilli et al. [19] have done a comprehensive analysis of different existing network forensic process models and frameworks. As a result, this study suggested a new improved and flexible Generic Process Model for network forensics.

This process model groups different phases into proactive and reactive processes regarding the real-time and post-attack forensic investigation scenarios. Therefore I take

this model as a basis for the VLAN hopping attacks investigation process because it provides both real-time and post-examination flexibility.

1.4 Scope and limitations

There are two VLAN tagging standards, the open IEEE 802.1Q and Cisco proprietary ISL. The study covers only the 802.1Q security issues, Inter-Switch Link (ISL) is excluded.

This study relies only on the Cisco hardware and technology which were only accessible equipment.

This study covers only switches that are operating at OSI Data Link layer. Multilayer switches are capable of routing, and therefore these devices are not traditional OSI Layer 2 devices and excluded.

The terms switch interface and switch port are treated as synonyms.

1.5 Acknowledgements

I would like to thank the following people: Mr. Lauri Võsandi (K-SPACE MTÜ), Mr. Luc Dandurand (Guardtime), Mr. Andres Septer (TUT) and Mr. Mati Leet for providing the testing hardware.

2 Data Link layer

Providing the detailed overview of each OSI layer or networking fundamentals is beyond the scope of the thesis. Because LAN switches are OSI layer 2 devices, it is necessary to cover most essential aspects of the Data Link layer and switch operation. The content is described only in the context of IEEE Ethernet standards.

There are two vendor-neutral networking models which are describing the open networking architecture [8, pp. 19–20]:

- OSI model. In the later 1970, the International Organization for Standardization (ISO) created the seven-layer Open Systems Interconnection (OSI) networking model.
- TCP/IP model. U.S. Department of Defense created the TCP/IP networking model.

Both models are referencing a comprehensive number of protocols that allow networked devices to communicate. The functions of the models are separated into different layers which are including standards and protocols to make the layers operational [8, pp. 20–21].

Throughout this thesis, the OSI reference model is used as a basis, because OSI model is much more flexible in separating different layers and has precise functions for each layer.

The original TCP/IP model defined single Link layer, but in a modern version of TCP/IP model, the Link layer is divided into Physical and Data Link layers to separate the physical bit based transmission from the encapsulation, de-encapsulation and addressing [8, p. 30].

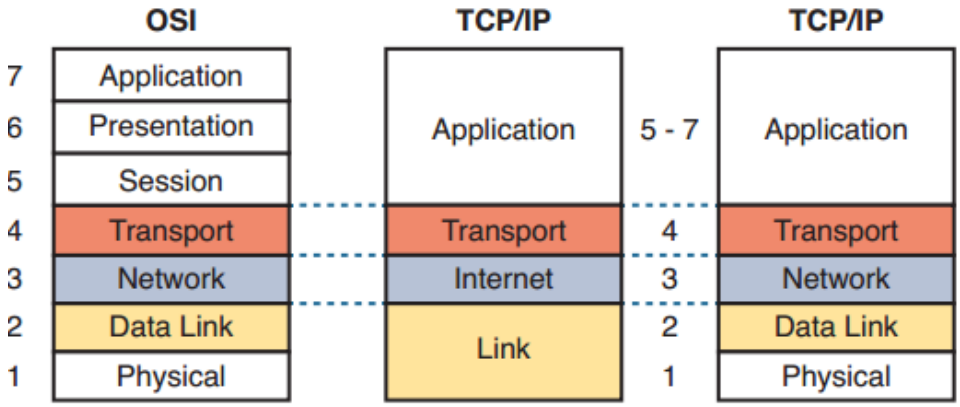

Figure 1: OSI model compared to the two TCP/IP models [8, p. 33].

Most commonly, the Physical layer can be represented throughout various IEEE Ethernet standards (e.g. 802.3, 802.3u etc.) which are determining the Unshielded Twisted Pair (UTP) physical links that are providing the transmission medium for the upper layers of the OSI model while encoding and sending bits over the wire. The Data Link layer is sending Ethernet frames from source to destination. The Ethernet frame contains the header and trailer of a data-link protocol including the data encapsulated inside that header and trailer [8, pp. 43–4].

As described in [20, p. 4], the primary data frame format that is specified by the IEEE 802.3 standard has the format with the following fields that are required by all Media Access Control (MAC) address implementations. MAC addresses are physical hardware addresses which are 6 bytes long. These are necessary for every port or device that connects to a LAN.

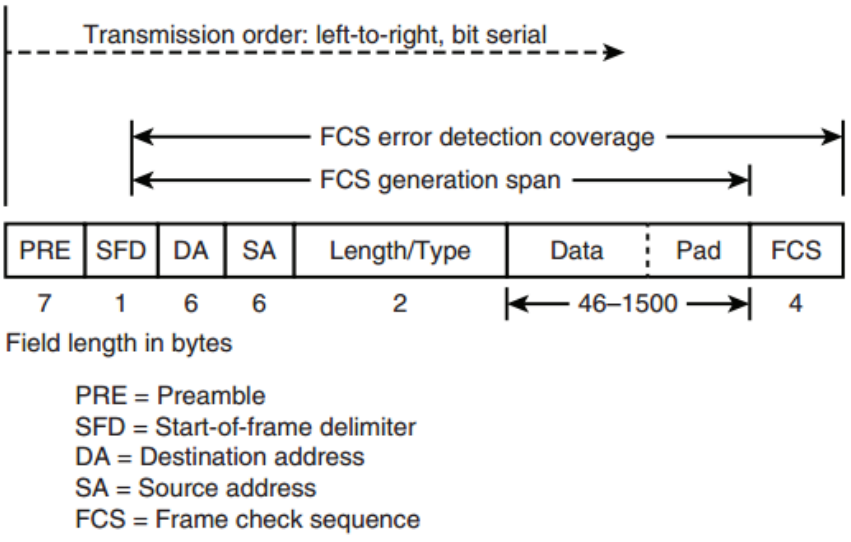

Figure 2: The basic IEEE 802.3 MAC data frame format [20, p. 4].

The structure of the Ethernet frame contains an Ethernet header at the beginning, the encapsulated data in the middle and a trailer at the end. Ethernet frame contains encapsulated data from the higher layers, usually IP packets. The padding is attached to satisfy the 46 bytes minimum length for this field [8, p. 52].

2.1 A Layer 2 Ethernet switch operation

An Ethernet switch is an OSI layer 2 device, which it is operating at the Data Link layer.

The general architecture of typical network devices usually contains three planes [20, p. 28] [8, p. 166]:

- Management plane - is responsible for management services like SSH, SNMP, Syslog etc. and managing the device itself.
- Control plane - responsible for protocols, processes and configurations that controlling the data plane operation or what the switch is doing.
- Data plane - responsible for actual layer 2 forwarding of frames that are generated by the devices that are connected to the switch. Data plane functions are the primary objective of the Ethernet switch.

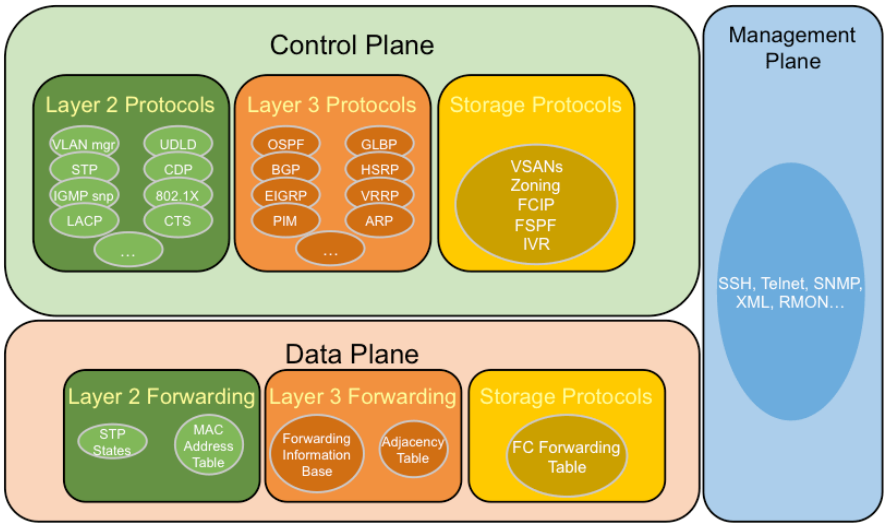

Figure 3: Management, Control and Data planes [21].

The picture above represents the comprehensive overview of the planes used in various networking devices including the underlying protocols and components.

The switch is utilizing destination MAC address of the received frame to transmit the frame to the next destination. The switch is listening to incoming traffic, collects Ethernet frames and learns the source MAC addresses, ports and VLAN IDs. If the received frame contains a new MAC address which is absent in the MAC address table, the MAC address, related port and VLAN ID are recorded in the forwarding table which is also called as a Content Addressable Memory (CAM) table. Usually, the switch is looking its MAC address table to determine the destination where frame must be sent. In case the MAC address of the frame is new and previously unknown, the switch is using a process called "unknown unicast flooding," where switch forwards the frame throughout all ports except the port the frame was received on. Multicast and broadcast frames are always flooded [20, p. 24].

The received frame is stored within the ingress queue of the switch port which can have multiple ingress queues at the same time. Ingress queues are used to apply predefined Quality of Service (QoS) or Access Control Lists (ACL) for the forwarding decision of the switch [20, p. 25].

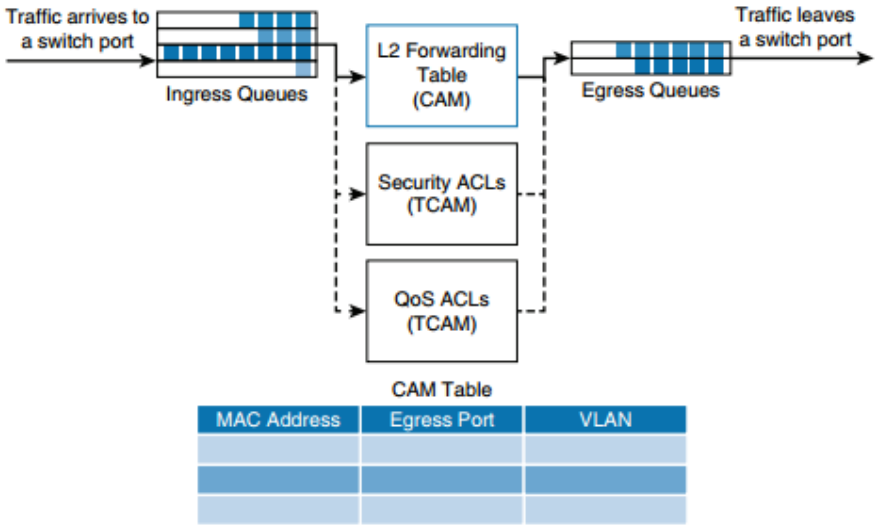

Figure 4: Layer 2 switch operation mechanics [20, p. 26].

As explained in [20, p. 26]: "Switches use specialized hardware to house the MAC table, ACL lookup data, and QoS lookup data. For the MAC table, switches use content-addressable memory (CAM), whereas the ACL and QoS tables are housed in

ternary content-addressable memory (TCAM). Both CAM and TCAM are extremely fast access and allow for line-rate switching performance. CAM supports only two results: 0 or 1. Therefore, CAM is useful for Layer 2 forwarding tables. TCAM provides three results: 0, 1, and don't care. TCAM is most useful for building tables for searching on longest matches, such as IP routing tables organized by IP prefixes. The TCAM table stores ACL, QoS, and other information generally associated with upper-layer processing."

To sum up, the OSI layer 2 switches are operating quite straightforwardly while processing and forwarding Ethernet frames.

2.2 VLANs and 802.1Q trunking

The VLANs are beneficial to create resilient network designs and reduce the number of devices that receive broadcast frames. VLANs also reduce security risks by reducing the number of hosts that receive copies of frames that the switches flood (broadcasts, multicasts, and unknown unicasts) and to improve security for hosts that send sensitive data by keeping those hosts on a separate VLAN [8, pp. 244–5].

IEEE 802.1Q is the open networking standard which implements supports virtual local area networks (VLANs) on an Ethernet network. The IEEE 802.1Q standard defines a method of VLAN tagging for Ethernet frames which is used by Ethernet switches. According to the IEEE 802.1Q, a 4-byte 802.1Q VLAN header inserted into the original Ethernet frame's header [8, p. 248].

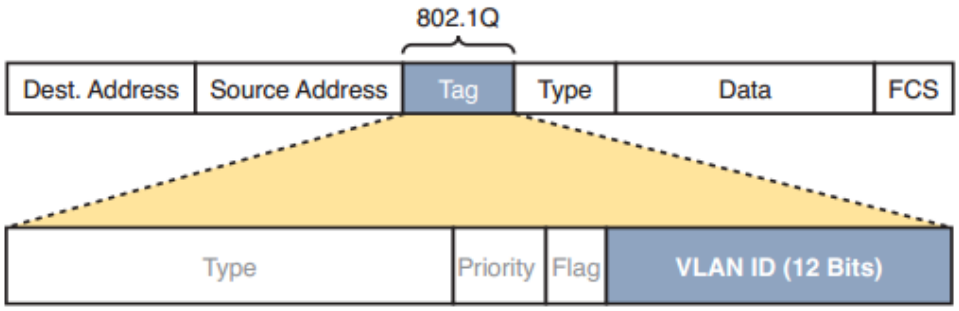

Figure 5: IEEE 802.1Q trunking [8, p. 248].

If VLANs implemented in networks that have multiple interconnected switches, those switches need to use VLAN trunking on the links between the switches. VLAN trunking lets the switches to use a method defined as VLAN tagging, by which the

sending switch attaches another header to the Ethernet frame before transmitting it over the trunk. This extra trunking header includes a VLAN identifier (VLAN ID) field, which is used by the sending and receiving switches to associate the frame with a particular VLAN ID. If the switch receives the frame, then it uses the VLAN ID of the frames for processing and forwarding [8, pp. 246–7].

VLAN trunking creates a link between switches that carry many VLANs. By default, the switches treat the trunk link as if it is a member of all the VLANs. At the same time, the trunk keeps the VLAN traffic logically separate, and frames in one VLAN would not go to devices in other VLAN because each frame is identified by VLAN ID as it passes over the trunk link [8, p. 248].

Trunk ports can communicate both 802.1Q tagged and untagged frames. The frames transmitted over a trunk link without any 802.1Q tags belong to the native VLAN. The primary function of the native VLAN is to allow a switch to use 802.1Q trunking. If the device on the other end of the trunk link does not support trunking (or the trunking is not enabled), the traffic of the native VLAN is transmitted over the trunk link, but the receiving switch is unable to process the frames [8, pp. 248–9].

By default, the Cisco switches are using the VLAN 1 as a default native VLAN where the 802.1Q tagging is dropped. If the receiving switch receives an untagged frame over a trunk port, and if that Ethernet frame is missing the 802.1Q tag, then the receiving switch assumes that the frame belongs to the native VLAN [22, pp. 20–1].

The default VLAN 1 is used by the switch to transfer other OSI layer 2 and Cisco proprietary protocol traffic, like Cisco Discovery Protocol (CDP), VLAN Trunking Protocol (VTP), Port Aggregation Protocol (PAgP), and (DTP) information. On Cisco environment, it is not allowed to delete the VLAN 1 from the switch [23].

2.3 Private VLANs

The architecture of Cisco Systems' Private VLANs is described in RFC 5517. Private VLAN (PVLANs) are designed for security, and they segment the VLAN further into several secured sub-domains [24].

Private VLANs are an advanced switch security feature that enables to create isolated or quarantined VLAN subdomains within the primary VLAN itself where the subdomain consists of a primary VLAN and one or more secondary VLANs [25].

PVLANs introduced the concepts of primary, secondary, isolated and community VLANs at OSI layer 2 device level.

As described in [25], the PVLAN can have only a single primary VLAN which forms the entire PVLAN domain. All the secondary VLANs are inside the single primary VLAN. Primary VLAN provide a special promiscuous port which can communicate with any other port inside isolated or community VLANs. There are two types of secondary VLANs within a primary VLAN:

- Isolated VLANs - all the switch ports which belong to the isolated VLAN cannot communicate with each other or any other switch port. A single isolated VLAN port can communicate only with the promiscuous port of the primary VLAN.
- Community VLANs - all switch ports inside single community VLAN can only communicate with each other or promiscuous port of the primary VLAN. Community ports cannot communicate with isolated VLANs, other community VLANs or any other switch port.

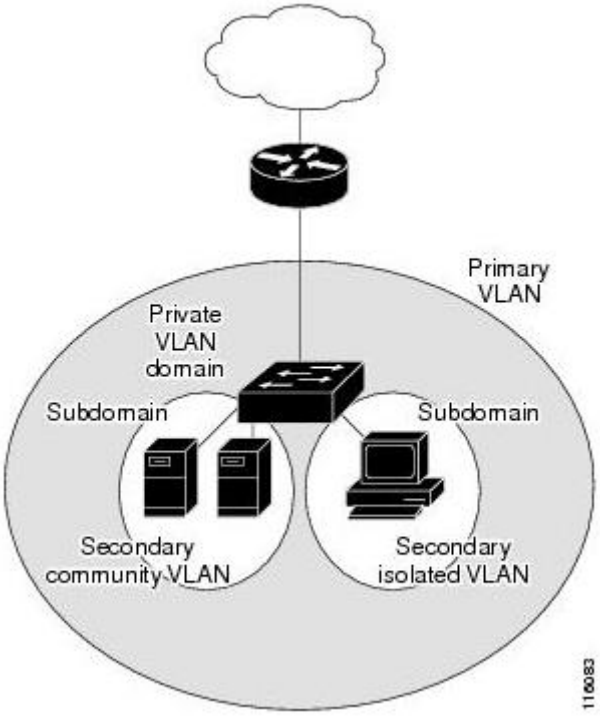

Figure 6: Private VLAN domain and secondary VLANs [25].

The traffic is strictly isolated at the switch level, but all isolated or community VLAN ports can communicate each other through the OSI layer 3 router port. The promiscuous port of the switch is directly connected to the router port [25].

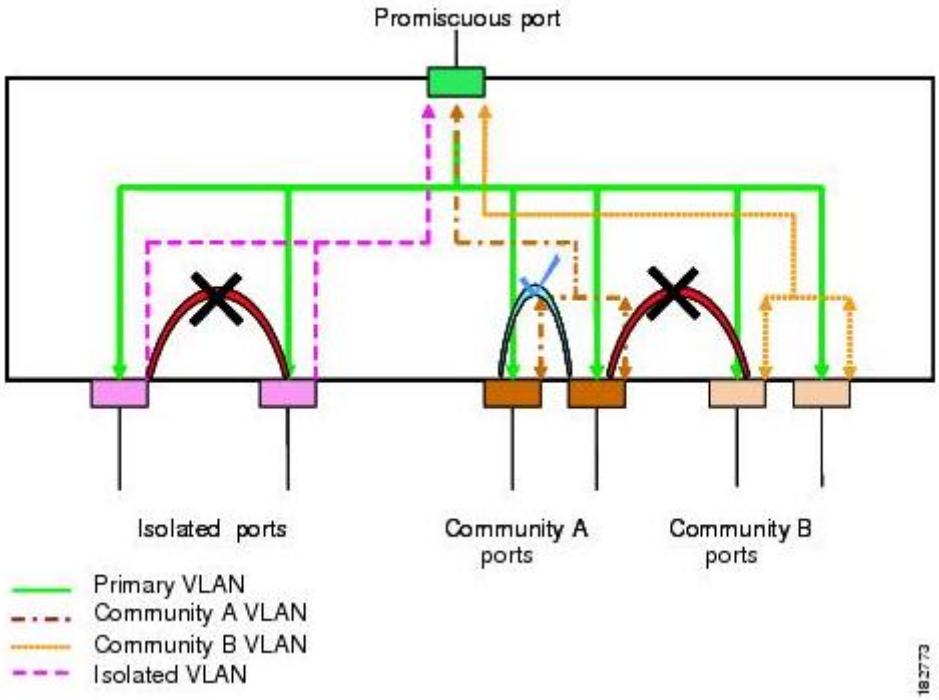

Figure 7: Private VLAN traffic flows [26].

The Figure above illustrates the Cisco's implementation of the private VLAN traffic flows through the promiscuous port of the primary VLAN. Isolated PVLAN ports can communicate only with the promiscuous port. Ports inside the community VLAN can communicate with each other or with the promiscuous port.

2.4 Voice VLAN

Voice over IP (VoIP) is a packet voice technology which allows voice calls over IP network.

In VoIP implementations, the actual analogue voice signals are converted and compressed to digital signals. Then the output is encoded by using some appropriate voice codec (coder/decoder). There are many proprietary and standardized codecs like G.711, G.729 and others. As a result, the encoded voice is fragmented into packets. During the packetisation process, the various protocol headers from different OSI layers encapsulated to encoded packets, for example, Internet Protocol (IP), User Datagram Protocol (UDP) or Real-time Transport Protocol (RTP) and so on. After the encapsulation, the packets are sent over the IP network to a destination where the [27, pp. 155–6].

To get the Cisco VoIP operational, the Cisco IP Phone which is connected to the switch needs a CME (Cisco Unified Communications Manager Express) capable router to provide the telephony services. The Cisco CME router is using the Skinny Client Control Protocol (SCCP) or the Session Initiation Protocol (SIP) to communicate the Cisco IP phone. SIP and SCCP are signaling protocols which allow communicating and controlling IP phones. The signaling messages make the phone to ring, after the incoming call is answered the actual audio communication stream is established by using the Real-time Transport Protocol (RTP) over UDP protocol [28, pp. 32–5].

There are many other signaling protocols, mostly H.323 family protocols (H.225, H.245, G711, G.729, T.120 and others) which are needed to establish VoIP calls and at the end to close the media streams [27].

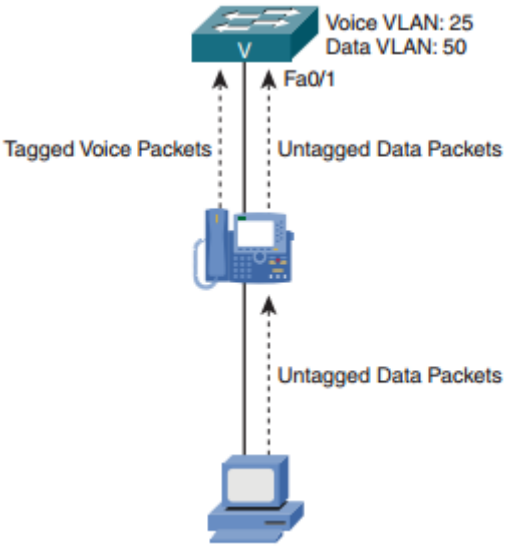

Figure 8: Separating voice and data traffic using VLANs [28, p. 58].

The IEEE 802.1Q tagging is supported by the Cisco VoIP infrastructure and IP phones. The IP phone establishes a trunk link between IP Phone and switch. This trunk link is using voice VLAN ID to tag packets. The switch interprets the VLAN tags and places the data in the correct destination VLAN. The data packets remain untagged and pass through the phone as usual data traffic [28, pp. 58–9].

In Cisco VoIP environments, the Cisco Discovery Protocol (CDP) is required to deliver voice VLAN information and configuration to the IP phone. After the IP phone receives the Voice VLAN ID, it starts tagging its packets. After the IP phone learns Voice VLAN ID and its IP address, then it starts to register itself in call processing server (CME) by sending its MAC address [28, pp. 61–5].

After registration, the IP phone needs the continuous connection to the call processing server. By default, the “keepalive” messages are sent after every 30 seconds. The absence of keepalive messages causes the keepalive failure [28].

3 Switch monitoring options

To monitor the behaviour of Ethernet switches, it is crucial to specify the different datasets that can be exported or extracted from these network devices. As shown by Santos in [29, p. 114], there are various sources for the data analytics: “Syslog, SNMP logs, server and host logs, packet captures, and files (such as executables, malware, exploits) which can be parsed, formatted, and combined with threat intelligence information and other network metadata to perform analytics."

It is crucial to collect all the appropriate data from Ethernet switch to perform the detection analysis of different VLAN hopping attacks. The acquired data may even overlap which makes possible to identify the correlations or other patterns and anomalies [30, pp. 23–4].

Modern business-class network equipment can generate numerous event logs which are sent to an external remote server via Syslog or SNMP because the network devices themselves have insufficient storage capacity [30, p. 305]. Ethernet switches usually support these data sources like Syslog, SNMP and Port Mirroring (SPAN) that can be aggregated by the external monitoring servers.

3.1 Syslog

Syslog is a widely used logging format, and a significant amount of network devices supports it, including most log management and analysis tools [31, p. 225]. The Syslog protocol (IETF RFC 3164) is usually acting as a client-server protocol and designed for transmitting event notifications over an IP network. Syslog protocol is using unreliable UDP port 514 for the data transmission and remote logging. It lacks any essential security features, no authentication or encryption. There are many other enhanced Syslog derivatives available like syslog-ng (next generation) and rsyslogd (reliable and extended syslogd) [30, pp. 297–8].

The logging messages are grouped by different sources that created them. This grouping is defined as a facility which can be any hardware device, particular network protocol, running system process, a software module or any another source [15]. The logging facility on Cisco IOS enables to store Syslog messages either locally or to send the

messages to the external remote logging server. By default, Cisco IOS is transferring logging messages to a logging process of the device which controls the transmission of logging messages to several destinations, such as the console or terminal output, logging buffer of the switch or external Syslog serve [32].

By default, Cisco IOS send log messages to console output for all severity levels of messages. When logging to the console or terminal is enabled, in case an event happens, the IOS transfers the messages automatically to console or terminal sessions, after the transfer the IOS can dismiss the messages. Cisco IOS has implemented two choices to store the copy of the old log messages. At first, the IOS can be configured to store the messages in memory buffer. The second option is to configure IOS to send log messages to external Syslog server [8, p. 781]. Buffered logs are volatile and will be lost after the power cycle and buffered logs may also be overwritten by IOS when available memory space is low [30, p. 353].

The system logging events are tagged with severity levels which range from debugging information at Syslog Level 7 to critical system emergencies at Syslog Level 0. The lower the number, the more severe the event that produced the message [8, p. 783].

Keyword	Numeral	Description	
Alert	0	Immediate action required	Severe
Emergency	1	System unusable	
Critical	2	Critical Event (Highest of 3)	Impactful
Error	3	Error Event (Middle of 3)	
Warning	4	Warning Event (Lowest of 3)	
Notification	5	Normal, More Important	Normal
Informational	6	Normal, Less Important	
Debug	7	Requested by User Debug	Debug

Figure 9: Syslog message severity levels by keyword and numeral [8, p. 783].

The severity levels of messages are a way to control the displayed messages. System logging is an excellent way to find out what was happening in a network device at the time an attack occurred, but Syslog itself has limitations. For example, it is necessary to collect logs from multiple endpoints, and in case the endpoint is compromised, it may not be reasonable to trust the logs that are coming from compromised sources [32].

Communicating logs across the network have various reliability issues, because logs may be dropped during the network congestions or outages. The regular Syslog is using unreliable connectionless UDP protocol, if the datagram is dropped in transition, the receiving server will have no record of it and the sending client will not know to retransmit. UDP datagrams are usually dropped when the receiving server is heavily overloaded with large volume of traffic. The other problem is related to the time synchronization, if the between multiple endpoints may differ which causes the integrity issues of event logs in transit. There are security concerns when the messages are transmitted in clear-text format, because the messages can be intercepted, read or altered in transit, the attacker may even inject fake event logs into network traffic [30, pp. 307–9].

3.2 Simple Network Management Protocol

Simple Network Management Protocol (SNMP) is a network protocol that provides a message format for communication between managers and agents. An SNMP manager is acting as a server by using special SNMP management software. These managers are also named as a Network Management Stations (NMS). SNMP agents, on the other hand, can be every kind of devices like printers, routers, switches etc. The software running inside these agents acknowledges the variables that make up various operational parameters, system statuses and counters of the device. This software inside the agents is named as a Management Information Base (MIB). Cisco IOS on switches and routers consist of the SNMP agent with built-in MIB database [33, p. 695]. Each MIB variable is determined as an object ID (OID) by the MIB. The OIDs are based on both the RFC standard and proprietary variables [33, p. 697].

The SNMP is a feature rich and flexible solution that supports various operations for network monitoring and active management. Regarding monitoring the SNMP can both push information from remote agents to central gathering server which is acting as an SNMP manager or poll networked devices from the central aggregation server. Additionally, the SNMP can be used to remotely control the configurations of remote devices [30, pp. 68–9].

NMS uses Get request messages (Get, GetNext, GetBulk) to request various information from the agents. Connected agents send Get-Response messages back to

NMS to provide the requested information. The NMS can also set management variables on agents with Set messages. SNMP agents can be configured to send notifications to NMS via Trap and Inform messages to list the state of specific MIB variables. The difference between SNMP Trap and Inform messages is related to the protocol mechanism. TRAP messages do not have reliability which means that if the SNMP agent sends the trap message out using the unreliable UDP protocol and the message gets lost in transit, then there is no error recovery mechanism. Inform messages are the same as Trap messages, but they have improved reliability features. Inform messages are also using unreliable UDP transport protocol, but Inform messages have added application layer reliability where NMS must acknowledge the receipt of the message to the sender. Trap messages are available since SNMP version 1 and Inform messages since SNMP version 2c [33, pp. 696–7].

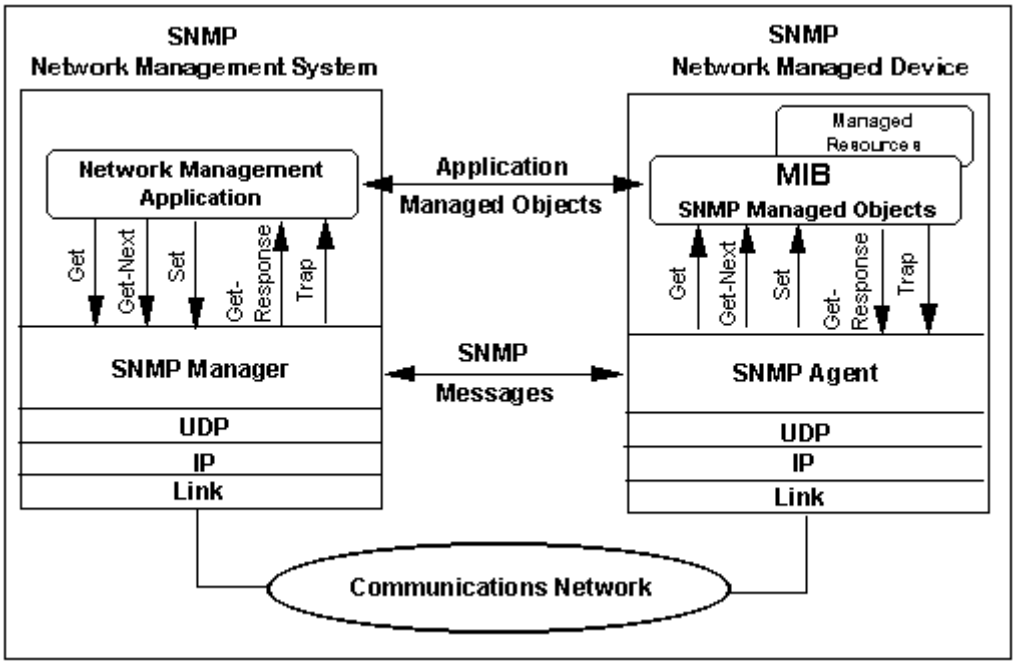

Figure 10: SNMP version 1 overview [34].

SNMP trap messages are a useful way to assure timely notifications about events happening on the device. SNMP version 1 and 2 are not secure, because these protocols are using plain-text community strings for authentication, which are sent across the network. Strong encryption is implemented since SNMP version 3 [30, p. 69].

3.3 Packet captures

The implementation of network analyser requires receiving traffic into its NIC which enables the analyser to capture and analyse the received frames. Configuring the port mirroring or SPAN on Cisco's Ethernet switches is a general way to capture traffic [33, p. 719].

Cisco switches are supporting various types of SPANs. The most general type is a local Switched Port Analyzer (SPAN) where all the monitored source and destination ports belong to the same switch. Another option is a Remote SPAN (RSPAN) where the source and destination switch ports do not belong to the corresponding switch. The third option is an Encapsulated Remote SPAN (ERSPAN) that supports source and destination ports on different switches. Only high-end Catalyst 6000 and 6500 series switches are supporting ERSPAN feature [35].

One local SPAN session can monitor single or multiple switch interfaces that can be monitored in both incoming (receiving or ingress) and outgoing (transmitting or egress) directions. SPAN is also supporting the multiple VLAN monitoring, but in this case, the SPAN is monitoring all the ports in particular VLAN (or group of VLANs), including the trunk ports. Each monitor session can use one or more SPAN sources of the same type - either switch port or VLAN. The switch does not treat the SPAN destination port as an ordinary switch port. The switch does not learn SPAN destination port MAC addresses for received frames or sent frames [33, pp. 720–22].

Trunk interfaces can also be configured as monitoring source ports. In this case, the SPAN includes frames from all VLANs on that trunk port by default, but SPAN VLAN filtering option can restrict the included VLANs [33, p. 721].

There are two critical port mirroring limitations that affect the quality of the traffic and deployment, one is scalability issue, and other is related to the prioritisation of SPAN traffic:

- The Port mirroring has a scalability issue which is inherently limited by the physical capacity of the switch itself. For example, as described in [30, p. 53]: "if you are using 100Mbps switch and you attempt to mirror four ports of it, which are each passing an average of 50Mbps to a single SPAN port. The total

amount of traffic from all four ports adds up to 200Mbps, which is far above the capacity of any one port to receive. The result is "oversubscription," the switch will drop packets."

- According to the Cisco's whitepaper [18]: "The switch treats SPAN data with a lower priority than regular port-to-port data. This means that, if any resource under load must choose between passing normal traffic and SPAN data, the SPAN loses and the mirrored frames are arbitrarily discarded. This rule applies to preserving network traffic in any situation. If there is not enough capacity for the remote SPAN traffic, the switch drops it. Knowing that the SPAN port arbitrarily drops traffic under specific load conditions, what strategy should users adopt so as not to miss frames? According to Cisco, the best strategy is to make decisions based on the traffic levels of the configuration and, when in doubt, to use the SPAN port only for relatively low-throughput situations."

These SPAN issues must be taken into consideration in terms of implementing the network analyser that must have access to network traffic for collection and inspection.

The port mirroring is not the only option for intercepting network traffic. An alternative option is to use and place inline network tap between networked devices. Inline tap is an OSI layer 1 device and they are using hardware to replicate data which leads to high fidelity packet captures. Another option for intercepting traffic is to use a network hub, which is also a layer 1 device where all connected devices share the same physical medium. Hub retransmits the received Ethernet frames to all other ports which enable to sniff the network traffic [30, pp. 47, 51–2].

The software is required to acquire and record the captured traffic from network interface adapter (NIC). Most common software packet capture libraries are libpcap on UNIX/LINUX and its port WinPcap on Windows systems. Popular packet sniffing and analysis tools like Wireshark, tcpdump etc. are based on libpcap libraries which can capture network traffic at the OSI Layer 2 level [30, pp. 54–5]. To capture all network traffic from the NIC, it should operate in a promiscuous mode which allows the NIC to receive all network traffic even if the traffic is not intended for it. If the NIC is not operating in promiscuous mode, the NIC compare MAC address of the received packet to its own MAC address. If the MAC address does not match, the packet is filtered [36, p. 314].

4 VLAN hopping attacks

As summarized in [4, p. 82] the OSI Data Link layer is open for large-scale of attacks, for example: “frame spoofing; frame forwarding; Spanning Tree Protocol injection; device removal or destruction; link unplugging; MAC flooding; MAC spoofing; root hijacking; root injection; topology refits; VLAN hopping; admin hijacking; filter injection; port mapping and disabling etc.” Other studies in [3] and [12] pointed out additional attacks against layer 2 including CDP attack, VLAN Trunking Protocol (VTP) attack, VLAN Management Policy Server (VMPS) attacks, Wireless 802.11 (Wi-Fi) attacks.

OSI layer 2 is also directly affected by the layer 1 attacks like signal jamming, eavesdropping or injections, damage of equipment or cables, which are passed to upper layers. A high level of intelligence and preparation is required for layer 2 attacks, because some attacks may put the network easily into an unstable state and halt the communication [4].

Each of these attacks forms a significant attack vector including tools, methods and best practice to successfully execute the particular attack. For example, this thesis covers only VLAN hopping attacks, but there are at least four different types of VLAN hopping attacks.

Covering other Layer 2 attack types, except VLAN hopping attacks, are beyond the scope of this thesis. These attacks were listed to illustrate the OSI layer 2 vulnerabilities. Also, VLAN management related VTP and VMPS attacks are excluded because these attacks are targeting the VLAN management protocols which are not related to VLAN hopping attacks.

4.1 VLAN security issues

In general, the VLAN hopping is an attack type that allows to sending traffic from one VLAN to another without using any OSI layer 3 device (router) while bypassing the logical VLAN segmentation [4], [12]. VLANs are configured on the switch level and VLAN hopping is happening inside the switch.

Unauthorised access to other VLAN without layer 3 device is violating the confidentiality criterion of the CIA triad principle. The three fundamental security control principles are confidentiality, integrity, and availability which is often called CIA triad. Confidentiality requires that the data not be available to unauthorised users. Integrity requires ensuring the data accuracy and completeness. Availability states that data, application or system must be available without impacting productivity [32, pp. 45–7].

VLAN vulnerabilities are not only related to Cisco devices. Common Vulnerabilities and Exposures (CVE) database search delivers 34 CVE entries that match the "VLAN" search term [37]. I also listed the CVSS score of each entry. As stated in [32] the Common Vulnerability Scoring System (CVSS) is a base metrics to analyse and classify the attack complexity which is a standard maintained by the Forum of Incident Response and Security Teams (FIRST).

Other vendors and devices are also affected in various forms. It is not necessary to list all the 34 CVE entries. The most interesting CVE entries are listed in the table below.

Table 1: Brief overview of CVE entries [37].

CVE-ID	**Description**	**CVSS Score**
CVE-1999-1129	"Cisco Catalyst 2900 Virtual LAN (VLAN) switches allow remote attackers to inject 802.1q frames into another VLAN by forging the VLAN identifier in the trunking tag."	7.5
CVE-2005-4440	"The 802.1q VLAN protocol allows remote attackers to bypass network segmentation and spoof VLAN traffic via a message with two 802.1q tags, which causes the second tag to be redirected from a downstream switch after the first tag has been stripped, as demonstrated by Yersinia, aka "double-tagging VLAN jumping attack."	5.0

CVE-2005-4441	“The PVLAN protocol allows remote attackers to bypass network segmentation and spoof PVLAN traffic via a PVLAN message with a target MAC address that is set to a gateway router, which causes the packet to be sent to the router, where the source MAC is modified, aka "Modification of the MAC spoofing PVLAN jumping attack," as demonstrated by pvlan.c.”	5.0
CVE-2011-0355	“Cisco Nexus 1000V Virtual Ethernet Module (VEM) 4.0(4) SV1(1) through SV1(3b), as used in VMware ESX 4.0 and 4.1 and ESXi 4.0 and 4.1, does not properly handle dropped packets, which allows guest OS users to cause a denial of service (ESX or ESXi host OS crash) by sending an 802.1Q tagged packet over an access vEthernet port, aka Cisco Bug ID CSCtj17451.”	7,8
CVE-2016-7039	“The IP stack in the Linux kernel through 4.8.2 allows remote attackers to cause a denial of service (stack consumption and panic) or possibly have unspecified other impact by triggering use of the GRO path for large crafted packets, as demonstrated by packets that contain only VLAN headers, a related issue to CVE-2016-8666.”	7.8
CVE-2016-2783	“Avaya Fabric Connect Virtual Services Platform (VSP) Operating System Software (VOSS) before 4.2.3.0 and 5.x before 5.0.1.0 does not properly handle VLAN and I-SIS indexes, which allows remote attackers to obtain unauthorized access via crafted Ethernet frames.”	10.0
CVE-2015-6675	“Siemens RUGGEDCOM ROS 3.8.0 through 4.1.x permanently enables the IP forwarding feature, which allows remote attackers to bypass a VLAN isolation protection mechanism via IP traffic.”	4.0
CVE-2013-4686	“The kernel in Juniper Junos 10.4 before 10.4R14, 11.4 before 11.4R8, 11.4X27 before 11.4X27.43, 12.1 before 12.1R6, 12.1X44 before 12.1X44-D20, 12.2 before 12.2R4, and 12.3 before 12.3R2, in certain VLAN configurations with unrestricted arp-resp and proxy-arp settings, allows remote attackers to cause a denial of service (device crash) via a crafted ARP request, aka PR 842091.”	7.1
CVE-2011-3593	“A certain Red Hat patch to the vlan_hwaccel_do_receive function in net/8021q/vlan_core.c in the Linux kernel 2.6.32 on Red Hat Enterprise Linux (RHEL) 6 allows remote attackers to cause a denial of service (system crash) via priority-tagged VLAN frames.”	5.7

The CVE analysis indicated that most common problems are related to software bugs and design flaws which can be grouped according to various levels:

- Firmware level;
- Linux kernel level;

- IEEE 802.1Q protocol design level;
- Protocol level e.g. DTP, CDP, VTP etc.;
- Feature design level, e.g. PVLAN.

Another study [13] researched the VLAN hopping attacks against virtualised switches implemented in Xen and Kernel-based Virtual Machine (KVM), VMware vSphere and Microsoft Hyper-V virtualization environments. The study found that most of the environments are vulnerable to Switch Spoofing and Double-Tagging attacks.

4.2 Switch Spoofing attack / basic VLAN hopping attack

Switch spoofing attack, also known as a basic VLAN hopping attack is an attack type that exploits the vulnerability in Cisco switches to take advantage of the Cisco's proprietary Dynamic Trunking Protocol (DTP) which is automatically negotiating trunk links between switches [13, pp. 2–3]. DTP is used to establish trunking on a link between two devices and it is also arranging the trunking encapsulation type which is commonly IEEE 802.1Q [12, p. 8].

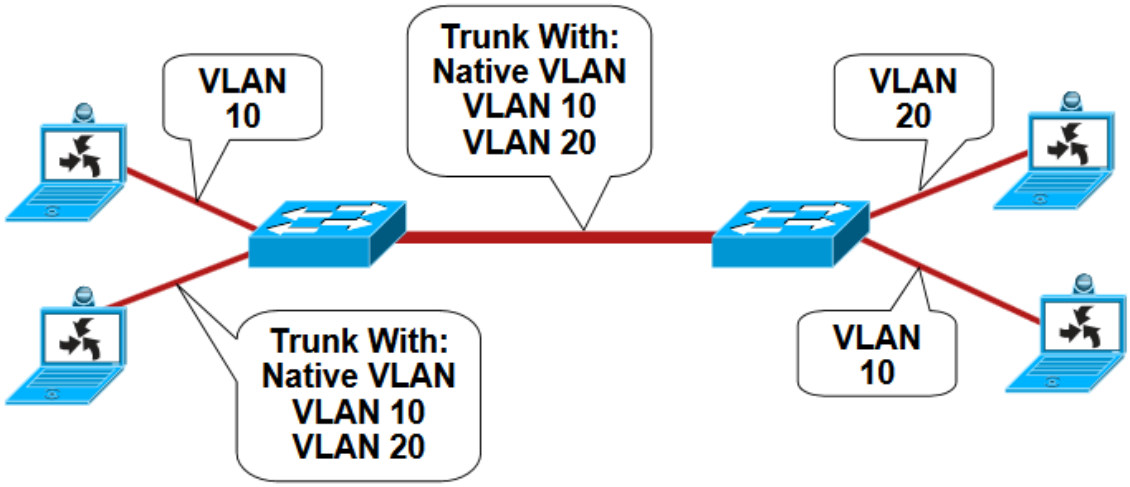

Figure 11: Basic VLAN hopping attack [38, p. 13].

Switches can use different DTP mode combinations to establish trunk links between each other. If the switches are both configured to "dynamic auto" mode, they will not be able to establish a trunk between them because the "dynamic auto" mode passively anticipates for the other side to initiate trunk link establishment. If two passively configured switch interfaces are connected, the trunk link will not be formed. The trunk

link will also form when one end is using DTP and the other end is manually configured to operate as a trunk port [23].

Table 2: The combinations of DTP modes [20, p. 54].

	Dynamic Auto	Dynamic Desirable	Trunk	Access
Dynamic Auto	Access	Trunk	Trunk	Access
Dynamic Desirable	Trunk	Trunk	Trunk	Access
Trunk	Trunk	Trunk	Trunk	Limited Connectivity
Access	Access	Access	Limited Connectivity	Access

By default, Cisco switches have DTP enabled on all ports and trunk links can easily be formed automatically. If physical ports on a Cisco switch are left for the default configurations, then an attacker can connect a system via any switch port and manipulate the switch into negotiating a trunk link. The trunk link is established after the successful attack which enables the attacker to gain access to all of the VLANs allowed in the trunk link including all the systems found on any accessible VLANs [13, pp. 2–3].

This type of VLAN hopping attack can be mitigated by shutting down unused switch ports to prohibit physical access to the switch ports. Another option is to disable the DTP protocol on switch ports to prevent any automatic trunk link negotiations [13, p. 4]. It is also recommended to configure switch ports as access ports to drop DTP frames [1, p. 101]. An additional option for protection is to apply unused VLAN to the unused switch ports [10, p. 16].

4.3 Double Encapsulated 802.1Q VLAN hopping attack

Double tagging VLAN hopping attack manipulates 802.1Q Ethernet frame tagging and tag processing mechanism of switches. Switches usually remove only one 802.1Q tag from the frame. In Double tagging attack, the original Ethernet frame is manipulated

and two VLAN tags added inside the frame. The malicious frame contains two VLAN tags (VLAN IDs). The IEEE 802.1Q specification does not deny multiple consecutive VLAN tags to be injected into one Ethernet frame, which enables the attacker to create the malicious Ethernet frame with two different 802.1Q tags [1, pp. 71–2].

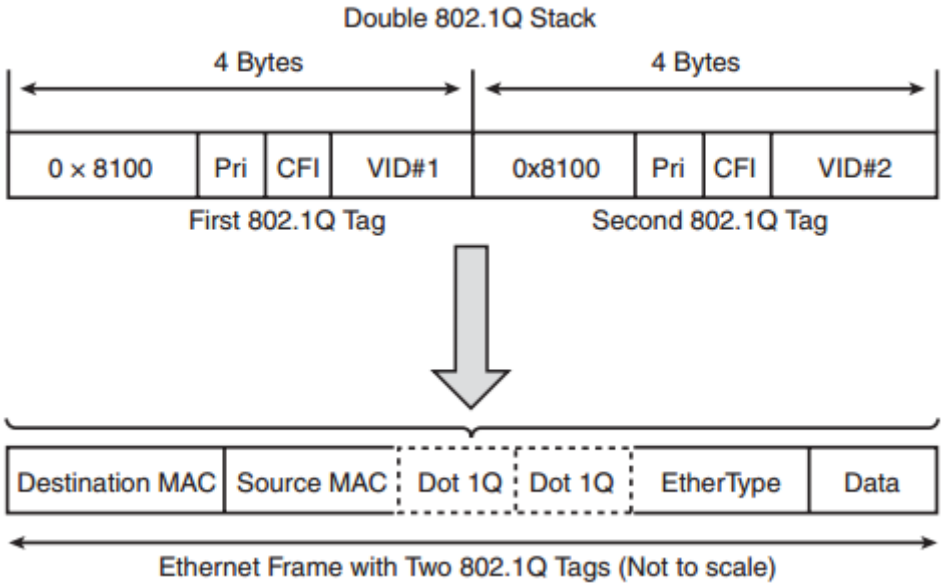

Figure 12: Ethernet frame with two IEEE 802.1Q tags [1, p. 71].

After the double encapsulated 802.1Q frame is transferred out from the attacker, the first receiving switch removes the first VLAN tag encapsulation. After that, the switch forwards the frame out with the second VLAN tag encapsulation. The second receiving switch removes the second VLAN tag encapsulation and transmits the frame to the host in the target VLAN. This type of VLAN attack is unidirectional [13, pp. 4–5].

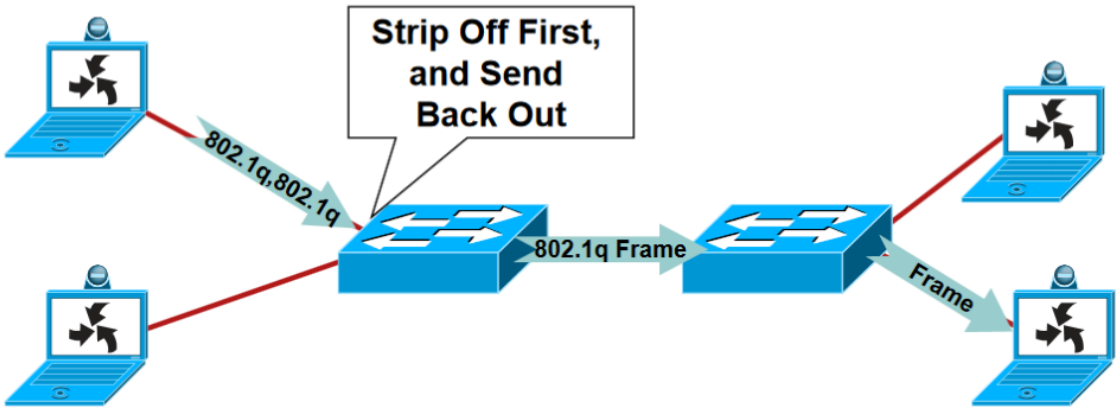

Figure 13: Double 802.1Q encapsulation VLAN hopping attack [38, p. 14].

This type of VLAN attack requires that the switch port of the target switch must have the same native VLAN ID as a VLAN assigned to trunk port between switches. For example, if the native VLAN of the trunk link is 1, then the compromised switch access port must also be in the VLAN 1 [12, p. 5].

This attack can be mitigated by configuring switch ports as access ports [10, p. 16]. Isolate and avoid to use default native VLAN for the switch ports [13, p. 6]. It is also reasonable to disable native VLAN from the trunk link and ensure that all traffic on the trunk is tagged [1, p. 74].

4.4 Private VLAN (PVLAN) hopping attack

Private VLANs (PVLANs) enable to create isolated virtual networks by limiting the ports which are allowed to communicate within the same PVLAN [5, p. 701].

A Private VLAN is protected from the layer 2 side, but is open to attacks from the layer 3 side. The whole idea behind the PVLAN attack is to manipulate the content of the Ethernet packet to bypass the intended Private VLAN security mechanism [12, pp. 10–11]. Therefore the PVLAN attack is using the expected behavior of a private VLAN against the private VLAN itself because the attacker can take advantage of the connected router to forward traffic to the switch promiscuous port of a PVLAN [39].

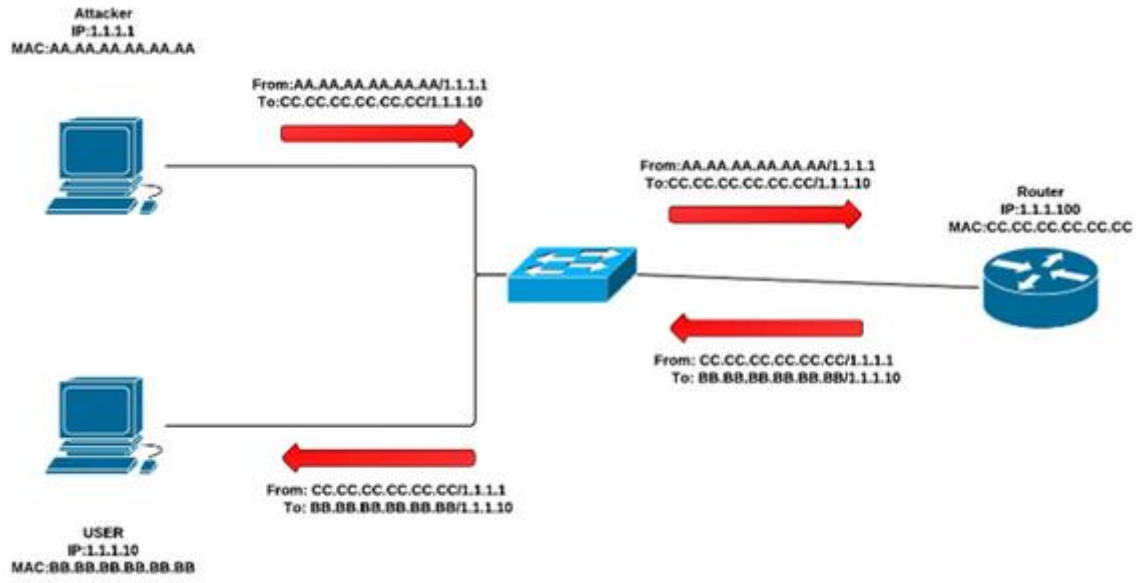

Figure 14: Private VLAN attack description [40].

The Figure above describes the private VLAN attack process. The Ethernet frame is manipulated in a way that the target's IP address and MAC address of the router's port are used [5, p. 701]. The router port must be linked to the promiscuous port of the switch which is used to forward the packet to the target host. The router rewrites the destination MAC address of the frame and forwards it to the target. This attack type is unidirectional, the attacker can only send traffic to the destination mechanism [12, pp. 10–11].

PVLAN hopping attacks can be mitigated by configuring an Access Control List (ACL) on the associated router interface or to use VLAN based ACL [5, p. 701], [12, pp. 11, 21–2].

4.5 Voice VLAN (VoIP) hopping attack

VoIP security alone is a big topic. VoIP systems deployed over the IP network which brings along all the vulnerabilities of IP network which includes the vulnerabilities found in VOIP applications, VoIP infrastructure (protocols) and network devices [41].

There are various vulnerabilities and attack vectors found in the VoIP protocol layers and VoIP is open to physical attacks against VoIP equipment like IP phones [42, p. 2].

Describing all the VoIP vulnerabilities is beyond the scope of this thesis which focuses on voice VLAN hopping.

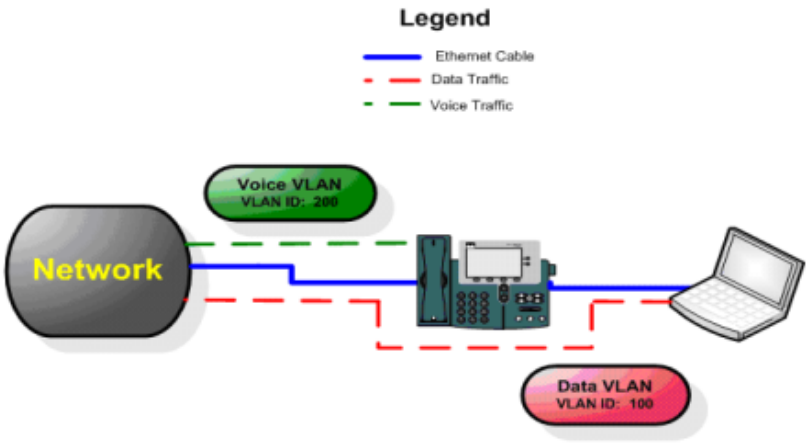

Figure 15: VoIP implementation [27, p. 158].

As described in chapter 2.4, the common VoIP implementation includes both data and voice VLANs. The idea behind voice VLAN hopping is to hop from voice VLAN to data VLAN or vice versa to get unauthorized access to particular network behind the VLAN. The attacker can use either IP phone alone or PC for VLAN hopping [43].

From voice to data VLAN hopping the attacker need to get physical access to the IP phone and pretending the behavior of IP phone. For a successful attack, the first step it to capture CDP packets to discover the voice VLAN ID from the traffic. The second step is to find available IP address of the target subnet. After that, the attacker needs to create the VLAN sub-interfaces to get unauthorized access to both data and voice VLANs [43].

Voice VLAN hopping attack is similar to Switch Spoofing Attack because both require creating sub-interfaces on the operating system to get unauthorized access to target VLAN. Recall from chapter 2.4 that the IP phone establishes a trunk link between IP phone and switch, where the IP phone using voice VLAN ID to tag its packets and data packets remain untagged. Because of the trunk link, both data and voice VLANs are accessible. After successful voice VLAN hopping the network is open for additional attacks.

Operational IP phones include both OSI layer 2 and 3 devices. Therefore it is necessary to apply hardening security countermeasures in terms of both OSI layers.

Various options are available to mitigate Voice VLAN hopping attack. It is recommended to use the firewall to separate voice and data subnets, enable MAC address filtering and 802.1X authentication [43]. Apply hardening configurations at CME level by disabling web access, access to PC voice VLAN etc. [44]. The attacker can directly use Ethernet cable of the IP phone. Therefore it is recommended to apply Cisco specific advanced security features like IP Source Guard (IPSG), Dynamic ARP Inspection (DAI) and DHCP snooping which help to prevent IP spoofing [45, pp. 99–100]. First of all, ensure the physical security of the IP phones and apply encryption if possible [46, p. 28].

5 Research model

The research involves repetitive actions to execute different VLAN hopping attacks to test various port mirroring configurations and Ethernet switch reactions. Therefore it is necessary to have a research model which helps to determine and provides guidelines for the whole research process.

Because the research involves the investigation of network related simulated attacks against the networking equipment itself, then some kind of network forensic process model should be most appropriate. After investigating various models in [47], the Generic Process Model for network forensics appeared most suitable.

According to Pilli et al. [19, p. 15], the fundamental difference between network security and network forensics is based on the objective where network security is dedicated to protecting systems against the attacks while network forensics focuses on recording evidence of the attack itself. Network forensics can be treated as an advanced phase of network security because it requires acquiring data from various sources. The network security is mainly concentrating on the constant monitoring process, but network forensic includes post-mortem investigation of the attack.

Pilli et al. [19] have done a comprehensive analysis of different existing network forensic process models and frameworks. As a result, he proposed a new improved and flexible Generic Process Model for network forensics.

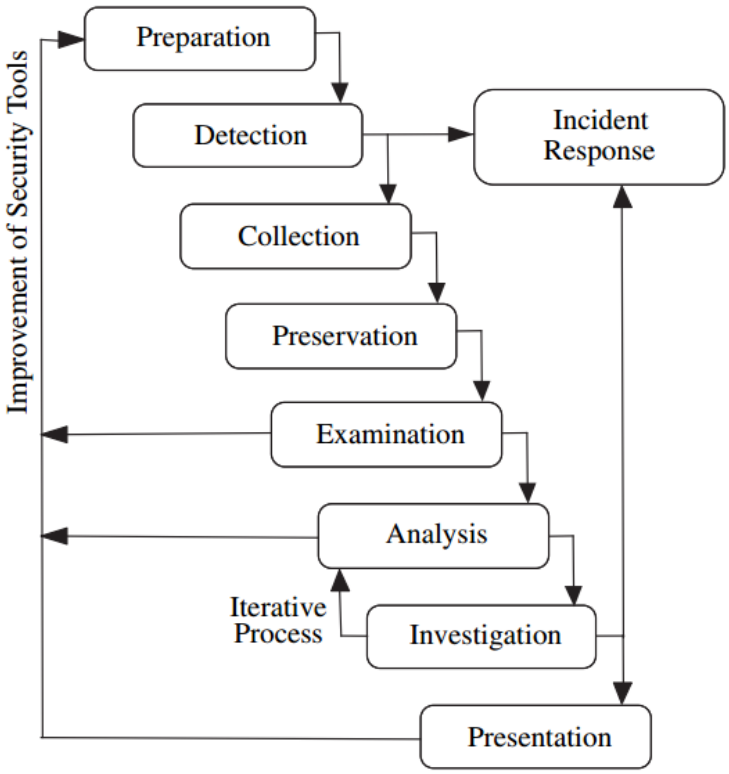

Figure 16: A Generic Process Model for network forensics [19, p. 20].

The model as described in [19] consist of several logically linked phases and the key the objectives of each phase are following:

- Preparation phase - all the network security monitoring tools and sensors are in place at various strategic points on the network.
- Detection phase - observe generated alerts and detected anomalies by various security tools and sensors. It is required to perform a fast preliminary validation to confirm the alert as an attack and not as a false alarm. The objective of the detection phase is to help the quick discovery of the attack.
- Incident response phase - the detected and validated alert or anomaly may immediately become an incident concerning security policy or legal basis of the organisation.
- Collection phase - all predefined sensors are collecting data (logs, alerts, traffic etc.). There should be transparent procedures, reliable hardware and software for data collection to ensure the quality of the evidence.
- Preservation phase - need to ensure that all the collected original data traces are hashed and stored on backup devices. Only copies of the data are used for the further investigation to preserve the original data integrity.

- Examination phase - the acquired data from different sources are integrated to form datasets for analysis. It is necessary to identify minimal attack attributes to locate the possible evidence. Also, need to deal issues like redundant information, overlapping time etc. to ensure the data quality. This phase may provide feedback to improve the tools to assist the network forensic process.
- Analysis phase - different analysis tools and approaches are used to research the data and match attack attributes and patterns. All the findings and data are analysed and correlated to understand the attack nature. This phase may provide feedback to improve the tools to assist the network forensic process.
- Investigation phase - the primary objective is to solve the attribution of the attack to track back to the point of attack origination and identify the attacker. This phase is iteratively performed with the analysis phase because investigation may require some additional features from the analysis phase.
- Presentation phase - the entire case and actions are documented, conclusions are presented understandably while providing explanations of the various procedures used to reach an outcome.

The Pilli et al. described in [19, p. 21] that the model is also separated into two groups of phases, the first five phases can handle the real-time network traffic, but the model is also suitable for the post-attack forensic investigation scenarios which begin at examination phase.

Another independent study [48] described and evaluated Generic Process Model for network forensic and pointed out that the first five phases are working proactively during the occurrence of the cybercrime which saves time and cost during the investigation process. The second group of phases which begins with examination phase act as a reactive process, they trace data and identify the attack indicators for the further analysis phase.

In addition to the linked phases, this grouping of different phases into proactive and reactive processes in terms of the real-time and post-attack forensic investigation scenarios provides additional flexibility. Therefore I take this model as a basis for the VLAN hopping attacks investigation process because it provides both real-time and post-examination flexibility. The model also defines the precise requirements for each phase.

6 Applying the research model to VLAN hopping study

In this chapter, I apply Generic Process Model for network forensics to research switch monitoring possibilities to detect different VLAN hopping attacks. Although the model is applied in the laboratory environment, the overall logic of the model is relevant for the forensics of simulated VLAN hopping attacks.

The first five proactive phases (preparation, incidence response, detection, collection and preservation) of the model are suitable to prepare the strategy for data collection, determine the strategic points of the sensors, perform the data collection process and preserve the collected data. To sum up, much preliminary work has to be done to get relevant data for the analysis.

The four remaining reactive post-attack phases (examination, analysis, investigation and presentation) are suitable for the detailed forensics research of different VLAN hopping attacks.

The effectiveness of the model is the continuous interaction between analysis and investigation phases which enable to continuously perform the forensic investigation cycle until the evidence is found or there are no remaining options left.

Because this model implemented in the controlled laboratory environment, it is not necessary to handle the incident response phase of the model, and therefore this component was excluded.

Furthermore, because the data is collected appropriately in the controlled laboratory environment, there are no data quality or preservation problems as described in preservation and examination phases of the model. Therefore these phases can be excluded.

During the investigation phase, the objective is to analyse detection and attribution possibilities of the different VLAN hopping attacks. Although I know the origin of

simulated attacks and therefore there are no attribution problems, it is necessary to get verification for this.

To use the forensics model for forensic research of the VLAN hopping attacks, I need to adjust the model by adapting it to handle the attack simulation process. I propose the following change in the model:

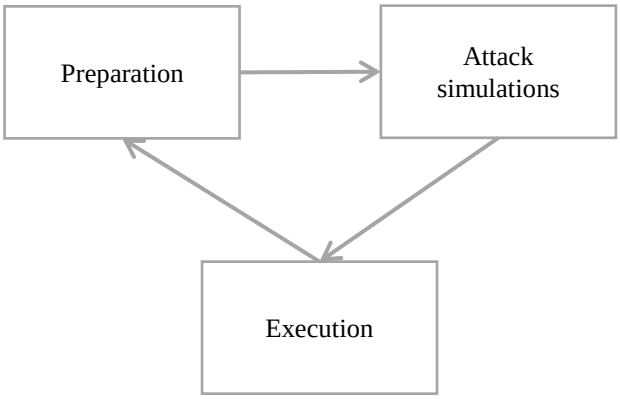

Figure 17: Proposed model change.

This change enables to use the model to continuously simulate different VLAN hopping attacks and collect the data for further forensic analysis. In this way, the forensic model can be used as a process-based model for laboratory experiment to research the detectable attributes and patterns of VLAN hopping attacks.

6.1 Preparation

The preparation phase requires that all the monitoring tools and sensors are in place at various strategic points on the network [19]. This requirement indicates that it is necessary to determine where to put the sensors for data collection (where are the strategic points for collection) and what kind of sensors can be used?

The objective of the thesis was to research how the various monitoring options of the switch can be used for the detection of VLAN hopping attacks?

On the assumption of this objective, it is also possible to treat single VLAN or group of VLANs as a strategic monitoring point and use SPAN feature for the data collection.

According to the IEEE 802.1Q and Cisco's switch technology, which I described in chapter 2, there can be various types of VLANs inside the switch:

- Default VLAN 1 – cannot be deleted or disabled, it is used by the CDP, VTP, PAgP, DTP and other OSI layer 2 control protocols. It is required for switch operation.
- Native VLAN – no 802.1Q tagging is applied, also necessary for trunking.
- Data VLAN - ordinary VLAN, which is tagged and can be used by both attacker and victim.
- Voice VLAN – special VLAN type for VoIP communication.
- Private VLANs – contains primary VLAN, secondary isolated and community VLANs.

In addition to VLANs, the switch has physical interfaces, which can also be configured to operate as Trunk interfaces. The Trunk interfaces can see all the traffic of different VLANs. Trunk interfaces are used to connect multiple switches together which contain different VLANs. It is necessary to treat trunk interface as a monitoring point to find any traces of VLAN hopping attacks.

By using different VLANs and trunk interfaces as monitoring points, the separation should help to analyse the VLAN hopping events in action.

The switch interfaces, trunk interfaces and configured VLANs can be treated as strategic monitoring points and can be monitored through the port mirroring (SPAN), which sends data to external traffic collector. The switch is configured to act as a Syslog agent and SNMP agent to send information to the external logging server.

Attacker and target are also treated as monitoring points. Attacker represents the source, and the target is the destination which helps to analyse what happened in the middle when the attack is directed throughout the switch.

6.2 Detection phase

As described by the Pilli et al. in [19] the objective of the detection phase is to process alerts from various security tools to quickly discover and validate the attack incidence.

The laboratory environment is used to research the detection possibilities of various simulated VLAN hopping attacks. Therefore all attacks are already detected regarding the detection phase requirements. The configuration of Ethernet switch was altered continuously to collect data samples. The following VLAN attack types, described in chapter 4, were simulated to research possible attack attributes and patterns:

- Switch Spoofing / Basic VLAN hopping attack.
- Double Encapsulated 802.1Q VLAN hopping attack.
- Private VLAN (PVLAN) attack.
- VoIP hopping / Voice VLAN attack.

The predefined SPAN monitoring options may have different capabilities to detect the VLAN hopping attacks. Mirroring different combinations of VLANs is a proposed method to investigate attributes and patterns of different VLAN hopping attacks.

The same applies to Syslog and SNMP because their detection and reaction capability is unknown. I expect that Syslog operating at debug level may detect the attack events or provide some log messages. If all the SNMP traps are enabled, I presume that it is possible that some traps are reacting to the VLAN hopping events.

While applying various SPAN options, it is necessary to test many different SPAN monitoring session configurations to discover the detection abilities. The same applies to SNMP and Syslog. The objective is to find out which Syslog levels can detect the attack and which SNMP traps are reacting.

6.3 Collection phase

According to Pilli et al. in [19], the data must be collected from various predefined strategic points on the network by using appropriate procedures, hardware and software to ensure the quality of evidence.

According to Liu [7, pp. 440–1]: the procedures for Cisco switch forensics focus on three key areas:

- Gaining access to the switch - gain access using the switch's Web administrative interface, a serial console cable, Telnet session or program such as Cisco Network Assistant (CNA).
- Collecting the data from the switch - capture and collect volatile and non-volatile information from the switch. Examine running configuration files, CAM tables, VLAN databases, memory information, debug interfaces etc.
- Collecting the data from the logging server - implement external Syslog and SNMP logging options of the switch.

Regarding forensics of network devices, it is crucial to make a distinction between volatile and non-volatile evidence. Most of the information generated by the network devices is volatile which will be lost after the reboot, loss of power or after the caches or memory were purged or overwritten. Non-volatile data is stored in flash memory and NVRAM, which contain IOS and a copy of the IOS, configuration files. Log files are usually treated as non-volatile evidence if external services are used for collection. Volatile evidence covers running configurations, CAM tables, stored packets in a buffer before they are transmitted, statistics, processor processes and other memory content, console logging and live captures of the network traffic [7, pp. 294, 306].

The switch can be monitored interactively over console connection to acquire information from CAM table and other volatile information from memory. The console connection allows a serial cable connection using a Cisco's rollover cable [7, p. 326]. In console logging the events are displayed in real-time on the CLI console, interrupting whatever interactive session was occurring there [30, p. 352]. As stated in [30, p. 337]: "For forensic investigators, the CAM table of an Ethernet switch can be very valuable, since it contains the MAC addresses of the network cards communicating on the local subnet. This table is very volatile and can change quickly, depending on network

activity. When an attacker is trying to sniff local network traffic, the CAM table often contains clear evidence of suspicious activity."

Investing CAM table and other volatile information of the switch is an essential additional information source to analyse what is happening inside the switch during the VLAN hopping event.

The Ethernet switch is configured to act as an SNMP agent and Syslog agent. Two attack types which are PVLAN hopping attack and Voice VLAN hopping attacks require a router for providing services to the switch. Therefore the router is also configured to act as an SNMP and Syslog agent in various PVLAN and Voice VLAN hopping scenarios.

Server for the data collection is prepared for SNMP and Syslog logging and to acquire network data from the SPAN sessions.

Two additional monitoring points are used for the data collection, one at the attacker host PC and another at the target host PC. The traffic data acquired from two additional monitoring points enable to analyse what happened between the source and destination.

Also, the console session is established to the switch to interactively acquired volatile data like CAM tables from memory.

6.4 Analysis and investigation phases

The analysis and investigation phases are closely tied. Most of the data is collected via port mirroring, and the primary analysis covers traffic analysis which should enable to identify the traces of VLAN hopping events in the traffic.

As described in [30, p. 75], the traffic analysis covers protocol analysis, packet analysis and stream analysis which enable to analyse fields within protocols, protocols within packets and packets within streams.

Protocol analysis is often needed together with packet analysis to interpret the communication structures or packets within streams accurately. It searches binary, hexadecimal or ASCII values, TCP or UDP port numbers [30, pp. 82–86].

Packet analysis examines content or metadata of packets. There are three conventional packet analysis techniques [30, pp. 99–101]:

- Pattern Matching - this method is matching specific values (strings) within the packet capture to locate packets of interest.
- Parsing Protocol Fields - this method is extracting the contents of protocol fields to locate packets of interest.
- Packet Filtering – this method separates packets based on the values of fields in protocol payload or metadata to locate packets of interest.

6.5 Presentation phase

The presentation phase is the final phase of the process to present research results. Here is the applied model of Generic Process Model for network forensics. The picture below summarises all the phases of the research process.

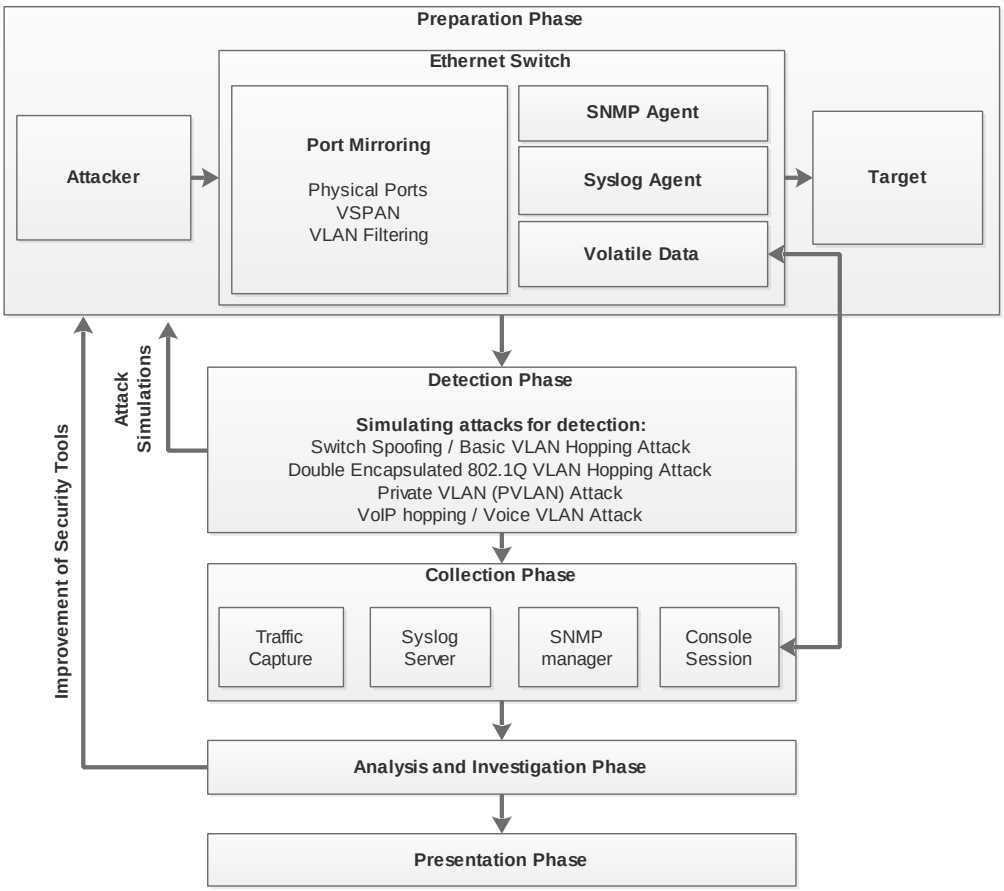

Figure 18: Applied Generic Process Model for network forensics.

The testing environment is prepared to simulate various attack scenarios and collect data which enables to analyse how various switch monitoring options are reacting to the different VLAN hopping attacks.

7 Testing environment

Before describing an actual testing environment, it is necessary to revise the testing objectives, network topology and main exploitation tools.

7.1 Testing objectives

According to the Cisco [17], if the local SPAN is used for the monitoring, then all the ingress packets are sent as untagged. By default, the SPAN feature of the switch does not monitor OSI layer 2 control protocols, such as VTP, DTP, CDP, STP, BPDU, PAgP etc. It is necessary to use corresponding configuration parameters on the SPAN destination port to acquire all traffic at OSI Layer 2 level. The configuration guide advice to use “encapsulation replicate” parameter on the destination port which should enable to monitor packets on all types including same encapsulation type or IEEE 802.1Q as they appeared on the source port.

The "encapsulation replicate" parameter is not supported by Remote SPAN (RSPAN) feature of the switch. The RSPAN strips off all VLAN tagging and excludes Layer 2 protocols. Therefore the RSPAN is not suitable for monitoring VLAN hopping attacks at OSI layer 2 level [17].

The local SPAN has several limitations and options for the data acquisition and special configuration parameters are required to monitor OSI layer 2 protocols to preserve source 802.1Q encapsulations.

The Ethernet packets might be dropped in various circumstances. For example, if the destination port is oversubscribed or the switch has congestion, the packets will be dropped. According to Cisco, it is possible that dropped ingress packets may appear on the SPAN destination or egress packets might be dropped from egress SPAN [17]. Switches may also drop the corrupted Ethernet packets [35]. Cisco has already stated that the SPAN traffic is treated with lower priority compared to the regular traffic of the switch [18].

Three different options are available to capture the network traffic from the switch [17]:

- Physical source ports. In this local SPAN monitoring method, all the traffic of the physical source port is monitored and mirrored. All types of ports are supported as source ports for monitoring – voice VLAN ports, trunk and access ports.
- VSPAN (VLAN based local SPAN) to monitor VLANs. Local SPAN supports to monitor specific VLAN or combination of VLANs. If VLANs are monitored as VSPAN sources, then all the traffic of active ports of the monitored VLANs are included and forwarded to the SPAN destination port.
- VLAN filtering. VLAN filtering is only supported by the trunk or voice VLAN ports. This method uses the local SPAN for monitoring trunk ports as SPAN source ports. Trunk ports can be configured to allow or disallow particular VLANs. If source trunk links are monitored, then all the allowed and active VLANs on the trunk link are forwarded to the SPAN destination port.

Private VLANs can be mirrored through VSPAN or physical source ports [25]. Typical used local SPAN configurations are described in appendix 3.

The objective of the thesis is to research how the various monitoring options of the switch can be used for the detection of VLAN hopping attacks?

VLAN hopping is happening inside the switch while the attacker is manipulating and sending malicious Ethernet frames to exploit various vulnerabilities of the switch. Is it possible that SPAN might drop these malformed frames because of several limitations?

Because the local SPAN supports multiple different options for the traffic acquisition, the objective is to research whether VLAN specific VSPAN or VLAN filtering options are more suitable and scalable for the VLAN hopping detection?

As explained in chapter 3.3, the conventional port mirroring of each physical port has scalability issues because it is difficult to mirror many switch ports at the same time or deploy many sniffers into the network.

Therefore VLAN specific VSPAN or VLAN filtering options could be more suitable and scalable because the particular VLANs or trunk links are mirrored directly.

Mirroring the physical ports enables to receive the ingress traffic, but VSPAN or VLAN filtering allows monitoring traffic at the VLAN level.

To sum up, the first objective is to find out which SPAN option is the most reliable or scalable to acquire the traffic to catch evidence of VLAN hopping events?

The second objective is to test the switch reactions against VLAN hopping events. The switch can act as a Syslog and SNMP agent to send log and trap messages to an external collector (server). What log messages are sent and which SNMP traps are reacting?

7.2 Network topology

The testing network was created to research the traces of VLAN hopping attacks and to answer the research questions. The following hardware and software were used for testing.

Table 3: Used hardware and software.

Hardware device	**OS Version**	**Special Software**
Cisco Catalyst 2950 switch	C2950-I6Q4L2-M Version 12.1(22)EA1	
Cisco Catalyst 2960 switch	C2960-LANBASEK9-M 15.0(2)SE7	
Cisco Catalyst 3560 switch	C3560-IPBASEK9-M 12.2(55)SE4	
Cisco 3660 Router	C3660-IK9O3S-M Version 12.4(18)	
Attack PC	Lubuntu Linux 16.04	Yersinia 0.8.2 VoIP Hopper 2.04 PackETH 1.6 Wireshark 2.4.5
Collector PC	Dualboot machine Lubuntu Linux 16.04 Windows 7	Wireshark 2.4.4 Tftpd server 4.52 PowerSNMP Free Manager 2.0 Putty; Minicom Cisco IP Communicator 8.6.6.0
Target PC	Windows 7	Wireshark 2.4.2 Cisco IP Communicator 8.6.6.0

Three Ethernet switches are used to provide more relevant results. The C2960 is used as a primary switch because it has a newest IOS version.

One Cisco 3660 router is added to the topology to make the PVLAN and VoIP services operational.

The PVLAN hopping attack requires C3560 switch which has a PVLAN support. Other two switches do not support PVLAN features.

Topology designed in a way that it enables to run and examine each attack scenario in isolation from other attack types as much as possible.

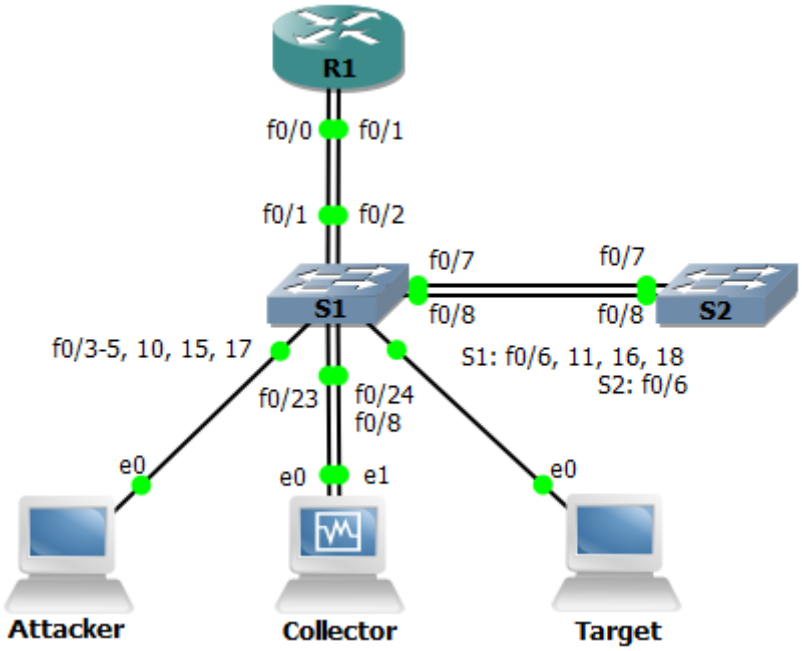

Figure 19: Network topology.

Switch ports f0/3, f0/4 and f0/5 are prepared for the various Switch Spoofing and Double Encapsulated VLAN Hopping attack scenarios. Port f0/3 left for default settings; port f0/4 moved for the VLAN 10 and f0/5 is configured as an access port in VLAN 10.

Switch S2 is needed for the trunk links between switches to mimic real conditions. Two trunk links used to test various SPAN configurations. Cisco IP Communicator software is used to verify the working VoIP functionality. A Syslog and SNMP version 2 is configured on the switch to send the log messages and SNMP traps to the collector which is acting both as Syslog server and SNMP manager.

7.2.1 Addressing table

Addressing table represents all the IP address ranges and used switch ports. Various scenarios require changing the switch ports of the attack and target PC-s.

Table 4: Addressing table.

Device	**Inter-face**	**IP Address**	**Subnet Mask (CIDR)**	**Default Gateway**	**Port (S1)**	**VLANs**	**Service**
R1	f0/0	n/a	n/a	n/a	f0/1	n/a	VoIP
	f0/0.10	192.168.10.1	/24	n/a	f0/1	VLAN 10	VoIP
	f0/0.15	192.168.15.1	/24	n/a	f0/1	VLAN 15	VoIP
	f0/1	10.0.0.1	/24	n/a	f0/2	N/A	PVLAN
R1	f0/0 or f0/1	192.168.99.15	/24	n/a	f0/1 or f0/2	VLAN 99	SNMP, Syslog for VoIP and PVLAN
S1	VLAN 99	192.168.99.10	/24	n/a	N/A	VLAN 99	SVI
S2	VLAN 99	192.168.99.20	/24	n/a	N/A	VLAN 99	SVI
S2	F0/7	n/a	n/a	n/a	f0/7	Trunking	Trunk
S2	F0/8	n/a	n/a	n/a	f0/8	Trunking	Trunk
Collec-tor	NIC 0	192.168.99.30	/24	n/a	f0/23	VLAN 99	Syslog SNMP
Collec-tor	NIC 1	192.168.99.31	/24	n/a	f0/24	VLAN 80	SPAN
Attack-er	NIC	192.169.1.5 Various	/24	Various	f0/3-5	VLAN 1 Various	Various
Target	NIC	172.16.20.20 Various	/24	Various	f0/6	VLAN 20 Various	Various
Phone1 (Attack er)	NIC	192.168.15.10	/24	192.168.10.1	f0/10	VLAN 10 VLAN 15	VoIP
Phone2	NIC	192.168.15.6	/24	192.168.10.1	f0/11	VLAN 10	VoIP

(Target)						VLAN 15	
Attack-er	NIC	10.0.0.100	/24	10.0.0.1	f0/15	PVLAN 200 Commu-nity	PVLAN
Target	NIC	10.0.0.10	/24	10.0.0.1	f0/16	PVLAN 200 Commu-nity	PVLAN
Attack-er	NIC	10.0.0.100	/24	10.0.0.1	f0/17	PVLAN 300 Isolated	PVLAN
Target	NIC	10.0.0.30	/24	10.0.0.1	f0/18	PVLAN 300 Isolated	PVLAN

* n/a indicates not applicable.

VLAN 1 is a default VLAN of the switch. VLAN 99 is representing the management VLAN for SNMP, Syslog and Switch Virtual Interfaces (SVI). VLAN 10 and 20 are representing data VLANs. VLAN 15 is representing voice VLAN. VLAN 100 is a primary VLAN of the PVLAN domain. VLAN 200 is a secondary community VLAN and VLAN 300 is a secondary isolated VLAN.

7.3 Attack tools

Three different attack tools are used to execute various VLAN hopping attacks. All the tools are available in Kali Linux distribution.

7.3.1 Yersinia

Yersinia [49] is an advanced network security and penetration testing tool which allows to execute a wide range of attacks against OSI layer 2 infrastructure. It is mainly designed to discover and test the vulnerabilities of the Cisco's proprietary network protocols, but many other IEEE open protocols are also supported. The following list expresses the supported OSI layer 2 protocols:

- IEEE Spanning Tree Protocol (STP);
- Cisco Discovery Protocol (CDP);
- Cisco Dynamic Trunking Protocol (DTP);

- Dynamic Host Configuration Protocol (DHCP);
- Cisco Hot Standby Router Protocol (HSRP);
- IEEE 802.1Q;
- IEEE 802.1X;
- Cisco Inter-Switch Link Protocol (ISL);
- Cisco VLAN Trunking Protocol (VTP).

The current version is 0.8.2 and it is released under a GPLv2 license, the tool is included in Kali Linux distribution [50]. Yersinia can be used to execute the two most common VLAN hopping attacks:

- Switch Spoofing or Basic VLAN hopping attack – it can be used to do take advantage of the DTP protocol vulnerabilities by sending the raw DTP packets to establish trunking.
- Double Encapsulated 802.1q VLAN hopping attack – It can taking advantage of the IEEE 802.1Q trunking protocol by sending RAW 802.1Q packet or sending double encapsulated 802.1Q packet.

Yersinia has three interface modes: command line (CLI), interactive mode and graphical mode.

7.3.2 PackETH

PackETH [51] is an Ethernet packet generator which allows to create and send wide range of possible packet types on the Ethernet link. PackETH supports the following protocols:

- Ethernet II, Ethernet 802.3, 802.1q, Q-in-Q, user defined Ethernet frame;
- ARP, IPv4, IPv6, user defined network layer payload;
- UDP, TCP, ICMP, ICMPv6, IGMP, user defined transport layer payload;
- RTP;
- JUMBO frames.

PackETH is an open source tool and it is provided under a GPLv2 license, it has both CLI and GUI interfaces. PackETH can be used to execute the PVLAN hopping attacks.

7.3.3 VoIP Hopper

The VoIP Hopper is a Linux command-line tool that is designed especially for the VoIP infrastructure security testing. It is released under a GPLv3 license. The tool has a built-in automatic MAC address spoofing features, if plugged and executed, the tool pretends the behavior of the IP phone [52].

It is included in Kali Linux distribution. The VoIP Hopper enables VLAN hopping into the voice VLAN on Ethernet switches. It is able to pretend the behavior of the IP Phone, in Cisco, Avaya, Nortel, and Alcatel-Lucent environments [53].

VoIP Hopper supports various network protocol automatic discovery like CDP, DHCP, LLDP-MED, 802.1q ARP, It also supports ARP sniffing and MAC address spoofing. The VoIP Hopper can automatically create a virtual VoIP Ethernet interface on the operating system and insert a spoofed 4-byte 802.1q VLAN header containing the 12 bit voice VLAN ID into a spoofed DHCP request [53].

7.4 Testing procedure

Previously, chapter 6 described the general research process which is based on the Generic Process Model for network forensics. This model was applied to the laboratory environment to provide general phases, guidelines and structure for the research process.

A more detailed procedure is needed to perform the action plan for testing. In the action plan, the chain of actions is presented to preserve the repeatability of the testing process.

The attacks are repeatedly executed to discover various SPAN session configurations and switch responses. The repeatable process of actions can be treated as a cycle. The following actions are executed during each testing cycle:

1. Clear logs and volatile data from the switch.
2. Verify the configurations and connectivity between endpoints.
3. Start data acquisition as described in chapter 6.3.
4. Execute attack according to specific attack scenario as described in chapter 4.

5. Verify the attack. The ICMP protocol is used for both bidirectional and unidirectional communication despite the fact that UDP protocol is more suitable for the unidirectional communication.
6. Explore volatile data produced by the switch; example commands are described in appendix 4.
7. Stop data acquisition software, preserve data.
8. Prepare the environment for the new scenario – change and apply new configurations on the hardware. The configurations are described in appendices;
9. Start at the beginning.

One switch testing scenario is used to isolate all possible influences. Initial configurations applied to get the necessary functionality working.

8 Attacks and results

According to the research objectives, multiple VLAN hopping attacks are simulated. This chapter covers testing results and findings.

8.1 Switch Spoofing attack / Basic VLAN hopping attack

If the switch port is not administratively shut down or the trunking mode dynamic negotiation parameter is set to auto or desirable, then the trunk link is formed between the attacker PC and switch port.

In Yersinia tool, the Dynamic Trunking Protocol feature enables the trunk formation between the attacker and switch.

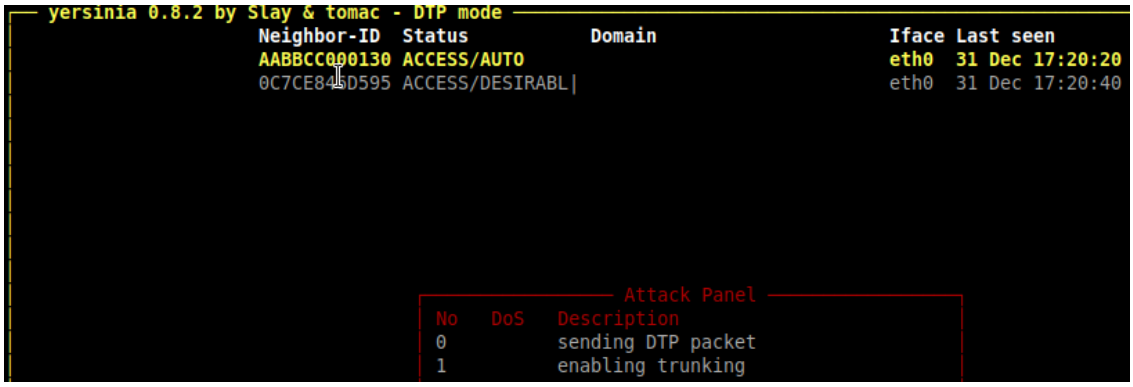

Figure 20: Establishing the trunk link.

After the VLAN sub-interfaces configured on the attacker operating system, all the reachable VLANs are bidirectionally accessible. This attack requires identifying the target VLAN IDs before creating the VLAN sub-interfaces. The VLAN IDs and IP address of the trunk link can be sniffed using the Yersinia's 802.1Q mode.

Table 5: The Switch Spoofing attack success states.

Source Switch Port	Source VLAN	Target Switch Port (S2)	Target VLAN	Successful
f0/3	1	f0/6	20	Yes
f0/4	10	f0/6	20	Yes
f0/5	10	f0/6	20	No

The establishment of the trunk link was unsuccessful only over the port f0/5 which was configured as an access port.

8.1.1 SNMP and Syslog results

The Syslog and SNMP agents configured on the Ethernet switch reacted to the VLAN hopping attack and provided outputs.

Table 6: An SNMP and Syslog reactions to Switch Spoofing attacks [54].

SNMP OID	OID Description	Syslog
1.3.6.1.2.1.17.0.2	"A topologyChange trap is sent by a bridge when any of its configured ports transitions from the Learning state to the Forwarding state, or from the Forwarding state to the Blocking state. The trap is not sent if a newRoot trap is sent for the same transition. Implementation of this trap is optional."	LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to down Line protocol on Interface FastEthernet0/3, changed state to up
1.3.6.1.6.3.1.1.5.3	"A linkDown trap signifies that the SNMP entity, acting in an agent role, has detected that the ifOperStatus object for one of its communication links is about to enter the down state from some other state (but not from the notPresent state). This other state is indicated by the included value of ifOperStatus."	
1.3.6.1.4.1.9.9.46.2.0.7	"AvlanTrunkPortDynamicStatusChange notification is generated by a device when the value of vlanTrunkPortDynamicStatus object has been changed."	
1.3.6.1.6.3.1.1.5.4	"A linkUp trap signifies that the SNMP entity, acting in an agent role, has detected that the ifOperStatus object for one of its communication links left the down state and transitioned into some other state (but not into the notPresent state). This other state is indicated by the included value of ifOperStatus."	

The SNMP trap provided clear OID 1.3.6.1.4.1.9.9.46.2.0.7 message which indicated the trunk link formation. Otherwise, the Syslog reaction was unreliable, the switch port changed down and back up without any other notification.

8.1.2 Traffic attributes

Based on the DTP and trunking functionality, it is reasonable to search the DTP and trunk formation related to the frame patterns in the traffic. There is no RFC available because the DTP is a Cisco proprietary protocol. The DTP frames and configuration states are easy to identify, and these can be treated as tokens that can be used for the

identification. The Following tokens can be easily identified from the traffic: Access/Auto; Access/Desirable; Trunk/Auto; Trunk/On; Access/Auto.

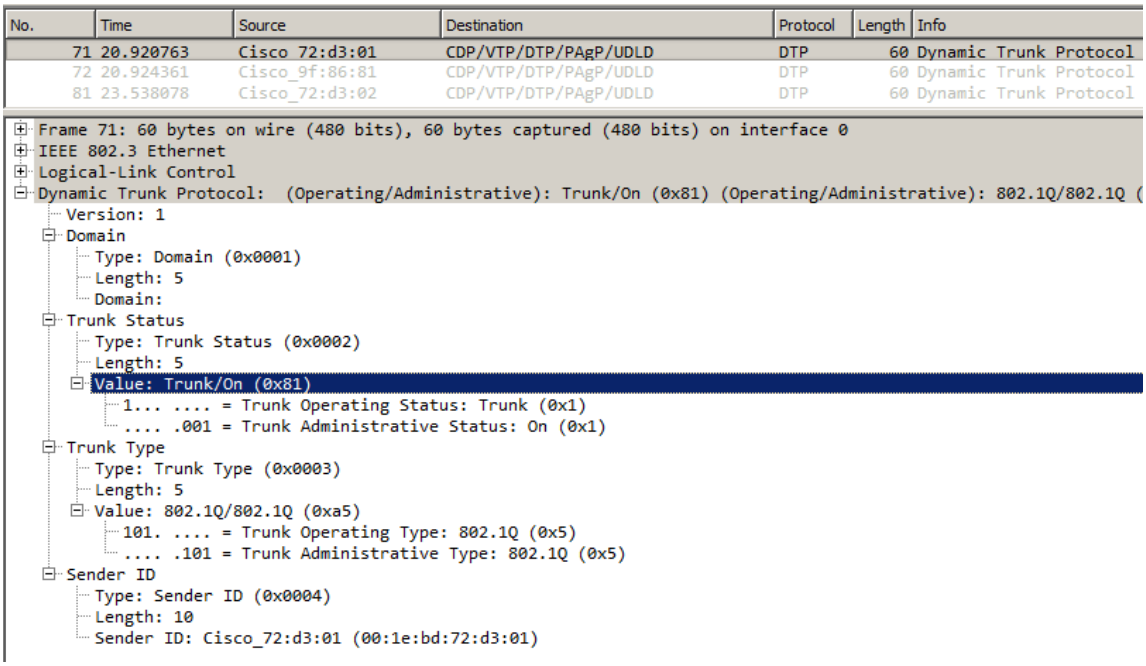

Figure 21: An example of captured DTP frame.

The Trunk/On is the token that follows after the successful trunk formation.

8.1.3 Compare packet capture methods

The testing resulted that in many cases when the attacker hopped from source to target VLAN, established the bidirectional connection and used ICMP ping to verify the connectivity, the VSPAN failed to process the packets if the VSPAN is monitoring only the single VLAN. The SPAN applied to the trunk port f0/7 failed to catch the VLAN hopping.

Table 7: SPAN and VSPAN results of the Switch Spoofing attacks.

Source Switch Port	**Source VLAN**	**Target VLAN**	**SPAN Source Configurations**	**DTP Tokens**	**Attacker Identified**
f0/3	VLAN 1	VLAN 20	f0/3	Yes	Yes
f0/3	VLAN 1	VLAN 20	f0/7	Yes	No
f0/3	VLAN 1	VLAN 20	VLAN 1	Yes	No
f0/3	VLAN 1	VLAN 20	VLAN 20	No	Yes
f0/4	VLAN 10	VLAN 20	f0/4	Yes	Yes

f0/4	VLAN 10	VLAN 20	VLAN 10	No	No
f0/4	VLAN 10	VLAN 20	VLAN 20	No	Yes
f0/4	VLAN 10	VLAN 20	f0/7	Yes	No
f0/3	VLAN1	VLAN 20	VLAN 1, VLAN 20	Yes	Yes
f0/4	VLAN 10	VLAN 20	VLAN 10, VLAN 20	Yes	Yes

Testing revealed many situations where VSPAN is configured to monitor the source VLAN, but VLAN hopping event enabled the attacker to jump to the target VLAN and become the member of the target VLAN which allowed to bypass the SPAN monitoring if the SPAN is not configured to monitor the target VLAN at the same time.

The situation can be explained further. As described in chapter 4.2, the attack requires creating sub-interfaces on the operating system to establish bidirectional communication between the endpoints. These sub-interfaces are configured as a member of the target VLAN. Therefore the communication bypassed the source VLAN which is monitored by the VSPAN.

For example, if the attacker hopped from VLAN 1 to VLAN 20 and the SPAN was configured to monitor only the VLAN 1, the hopping event caused traffic to bypass the source VLAN because the new channel established between sub-interface participating in VLAN 20 and target host which belonged to VLAN 20.

The next scenario used SPAN to monitor the trunk port of the switch. The VLAN filtering was applied to the trunk port f0/7. This method failed completely to discover the VLAN hopping.

Table 8: VLAN filtering results of the Switch Spoofing attacks.

Source Switch Port	**Source VLAN**	**Target VLAN**	**Allowed VLANs**	**DTP Tokens**	**Attacker Identified**
f0/3	VLAN 1	VLAN 20	VLAN 1	Yes	No
f0/3	VLAN 1	VLAN 20	VLAN 20	No	No
f0/4	VLAN 10	VLAN 20	VLAN 10	No	No
f0/4	VLAN 10	VLAN 20	VLAN 20	No	No
f0/3	VLAN 1	VLAN 20	VLAN 1, VLAN 20	Yes	No
f0/4	VLAN 10	VLAN 20	VLAN 10, VLAN 20	No	No

The functionality of the trunk link is described in chapter 2.2. Trunk links can transfer all VLANs, including tagged and untagged VLANs.

Testing revealed that SPAN failed to manage VLAN filtering of the trunk link if it is under VLAN hopping attack and debugging failed to provide any error messages.

8.2 Double Encapsulated 802.1Q VLAN hopping attack

This attack was most challenging to execute. Three different switches: Cisco Catalyst 2950, 2960 and 3560 used to execute the attack.

This attack failed entirely on Catalyst 2960 and 3560 switches because the double-tagged frames were stripped off or dropped by the switches even before the frames processed by the SPAN. Only the Catalyst 2950 switch processed the double encapsulated frame as expected.

As described in chapter 4.3, this attack requires two switches and the trunk link between the switches must have the same native VLAN ID as a VLAN assigned to an attacker's port.

Testing approved that the single switch failed to deliver the malicious frame to the host on destination VLAN 20. Therefore two switch scenario established and two linked switches C2950 (S1) and C2960 (S2) used for testing.

Table 9: SPAN results of the Double Encapsulated VLAN hopping attacks.

Source Port on S1	**Source VLAN on S1**	**Target VLAN on S2**	**Native VLAN of the Trunk Link between S1 and S2**	**SPAN source Configuration on S1**	**Attacker Identified**	**Attack Successful**
f0/3	VLAN1	VLAN 20	VLAN 1	f0/3	Yes	Yes
f0/3	VLAN1	VLAN 20	VLAN 50	f0/3	Yes	No
f0/4	VLAN10	VLAN 20	VLAN 1	f0/4	Yes	No
F0/5	VLAN10	VLAN 20	VLAN 1	f0/5	No	No

Catalyst 2950 switch forwarded the malicious frame to the target VLAN 20, and the frame appeared on the target host as expected.

This attack only worked when the attacker used the source port f0/3 with default configurations without any changes. By default, the default VLAN assigned to the port is VLAN 1 which is also configured as a native VLAN of the port. The attack failed if there were any changes from the default configuration.

The port f0/5 was configured as an access port which does not allow VLAN tagging except Voice VLANs. The switch dropped the double-tagged packets sent to the f0/5.

The two switch scenario revealed that the SPAN was able to catch the attacker, but the attack itself was unsuccessful. The second switch was unable to deliver the frame to target, but the SPAN on the first switch caught the malicious frame.

The main drawback of this attack is the unidirectional communication between the attacker and target host.

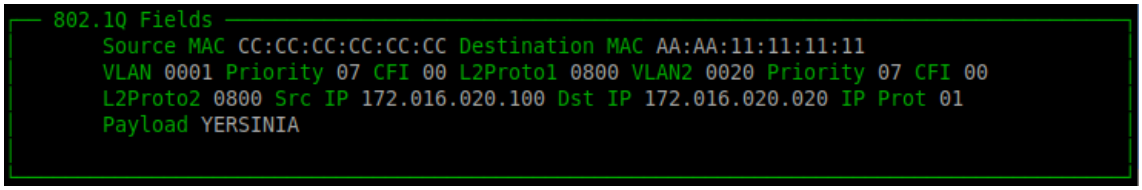

Figure 22: Editing 802.1Q fields in Yersinia.

In Yersinia, the 802.1Q field parameters enable to prepare the 802.1Q double- encapsulated frame. The target VLAN ID is needed to send the frame to the target host. As an example above, the following frame was sent to the target host.

8.2.1 SNMP and Syslog results

This attack scenario required two switches to execute the attack. The SNMP and Syslog configured on both switches S1 and S2. An SNMP and Syslog failed to provide any reaction to the Double Encapsulated VLAN hopping attack.

The scenario is tested on three switches and only Catalyst 2950 was able to process the malicious frame and treated it as ordinary network traffic. Catalyst 2960 and 3560 switches dropped the malicious frame without any SNMP or Syslog notification.

8.2.2 Traffic attributes

According to the theory, the first VLAN tag (ID 1) was stripped off by the switch as it appeared in the switch ingress interface (f0/3) and the second tag was stripped off on the egress interface (f0/6) while the frame forwarded to the host on target VLAN 20.

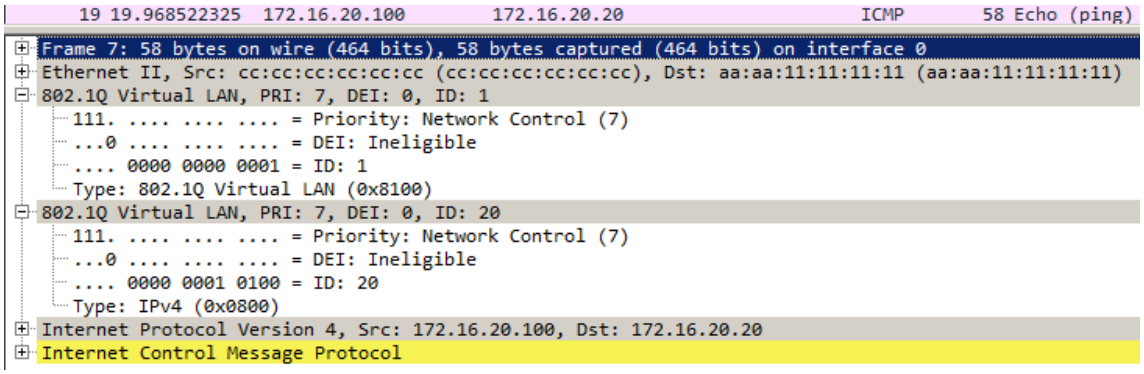

Figure 23: An example of the double-encapsulated frame on the attacker PC.

The frame processed by the Catalyst 2950 switch which stripped off the first tag (VLAN ID 1) and forwarded the frame to the target VLAN 20.

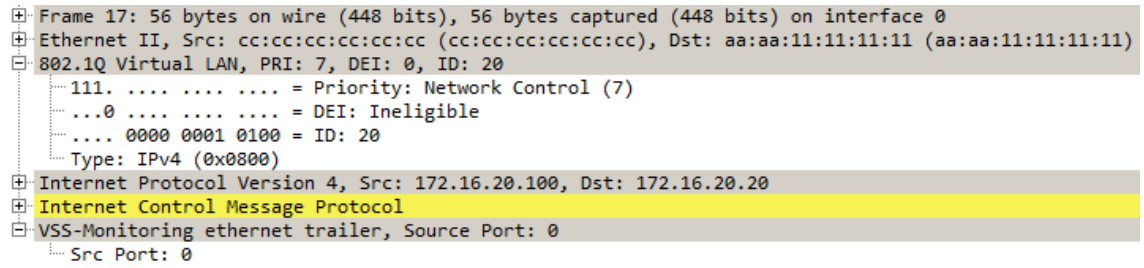

Figure 24: The double-encapsulated frame after the first VLAN tag is stripped off.

All the VLAN tags were stripped off by the switch after the frame appeared on the target host.

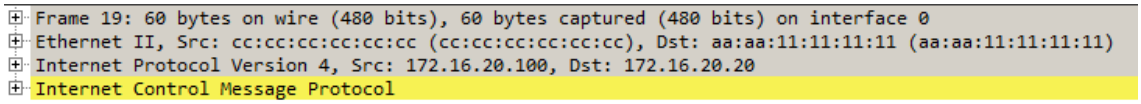

Figure 25: The double-encapsulated frame at the final destination.

8.2.3 Compare packet capture methods

This attack only worked on the Catalyst 2950 Switch with older version IOS version 12.1 which does not support the VLAN filtering features of the SPAN. Therefore the SPAN testing was limited to attacker's source port mirroring which was successful.

IOS version 12.1 supports the VLAN filtering of the trunk port. Because the trunk link on f0/7 used for the frame delivery, additional trunk link on port f0/8 established for the SPAN source mirroring.

Table 10: VLAN filtering results of the Double Encapsulated VLAN hopping attacks.

Source Switch Port	Source VLAN	Target VLAN	Allowed VLANs	Attacker Identified
f0/3	VLAN 1	VLAN 20	f0/7, f0/8	Yes
f0/3	VLAN 1	VLAN 20	VLAN 1 (f0/8)	Yes
f0/3	VLAN 1	VLAN 20	VLAN 20 (f0/8)	No

Testing revealed that if the attacker sent a malicious frame from VLAN 1 to VLAN 20, the SPAN which is monitoring VLAN 20 over the trunk link failed to detect the attacker.

8.3 Private VLAN (PVLAN) hopping attacks

According to the attack description in chapter 4.4, the private VLAN attack requires manipulating the destination MAC address inside the Ethernet frame. Therefore this kind of attack can be done using any Ethernet packet generation tool.

As an example, the PackETH tool which is described in chapter 7.3.2 is used for the Ethernet packet manipulation.

According to the PVLAN architecture which is described in chapter 2.3, two different attack scenarios can be executed:

- Isolated PVLAN inside attack – is it possible to reach to other host inside the isolated PVLAN?
- Attack between the community and isolated PVLANs – is it possible to reach to another host inside the isolated PVLAN?

According to the theory, both PVLAN attacks require that all attackable hosts in PVLANs must be connected to the same promiscuous port on switch S1.

8.3.1 Inside isolated PVLAN attacks

The hosts inside the isolated private VLAN 300 are restricted to interact with each other. All the communication is happening through the promiscuous port f0/2 on S1.

The hosts that are outside of the isolated PVLAN cannot connect to the hosts that are inside the isolated PVLAN. This attack requires the connectivity to the default gateway

on R1 and between the promiscuous port on S1. To satisfy this condition I had to make a change to the network topology and connect the Attacker PC to the previously configured switch port f0/17.

The addressing table below is representing the final topology change of isolated PVLAN 300.

Table 11: The isolated PVLAN testing topology.

Device	**Interface**	**IP Address**	**Subnet Mask**	**Default Gateway**	**Switch Port**	**VLANs**
Attacker	NIC	10.0.0.100	255.255.255.0	10.0.0.1	f0/17	PVLAN 300 Isolated
Target	NIC	10.0.0.30	255.255.255.0	10.0.0.1	f0/18	PVLAN300 Isolated

According to the PVLAN restrictions, the Attacker who is inside the isolated PVLAN cannot connect to the target PC because the isolated PVLAN allows only the communication to the promiscuous port f0/2 on S1.

The attacker needs to identify the target's IP address and MAC address of the port on the router R1 that is linked to the promiscuous port on the switch S1.

The inside attacker needs to manipulate the frame to change both the destination MAC and IP addresses in a way that a new MAC address is the router's interface MAC address and destination IP address is the target's IP address.

Table 12: An example of the modified malicious frame.

Frame	**Source MAC**	**Destination MAC**	**Source IP**	**Destination IP**
Original	aa:aa:aa:aa:aa:aa	aa:aa:11:11:11:11	10.0.0.100	10.0.0.30
Modified	aa:aa:aa:aa:aa:aa	00:06:53:4b:29:41	10.0.0.100	10.0.0.30

This type of attack is unidirectional, the attack was successful and the target host received the frame.

This attack allows the attacker to use any IP and MAC addresses as a source address to reach the destination, but cannot change the destination MAC address to send out the frame.

8.3.2 Attacks between the community and isolated PVLANs

In the second PVLAN attack scenario, the malicious connectivity tested between the community and isolated PVLANs. The objective is to test whether the attacker can connect from community PVLAN to the isolated PVLAN. The attacker is relocated from switch port f0/17 to f0/15 to execute the attack.

Table 13: The community and isolated PVLAN testing topology.

Device	Interface	IP Address	Subnet Mask	Default Gateway	Switch Port	VLANs
Attacker	NIC	10.0.0.100	255.255.255.0	10.0.0.1	f0/15	PVLAN 200 Community
Target	NIC	10.0.0.30	255.255.255.0	10.0.0.1	f0/18	PVLAN 300 Isolated

The second PVLAN attack scenario proved to be successful and allowed to send malicious packets between the hosts on the isolated and community PVLANs.

8.3.3 SNMP and Syslog results

An SNMP and Syslog are configured on the switch S1 and the router R1 to detect the possible reactions. Both PVLAN hopping attack scenarios remained invisible for the SNMP and Syslog. Both switch and router treated the malicious frame as regular frame and processed it appropriately.

8.3.4 Compare packet capture methods

After testing various local SPAN configurations to capture malicious traffic, shows that unhardened PVLAN architecture is much more vulnerable compared to the 8.2 Double Encapsulated VLAN hopping attack.

Table 14: SPAN and VSPAN results of the PVLAN hopping attacks.

Source Switch Port	Source VLAN	Target VLAN	SPAN Source Configurations	Attacker Identified
f0/17	PVLAN 300 Isolated	PVLAN 300 Isolated	f0/17	Yes
f0/17	PVLAN 300 Isolated	PVLAN 300 Isolated	PVLAN 300	Yes

f0/17	PVLAN 300 Isolated	PVLAN 300 Isolated	PVLAN 100	Yes
f0/15	PVLAN 200 Community	PVLAN 300 Isolated	f0/15	Yes
f0/15	PVLAN 200 Community	PVLAN 300 Isolated	PVLAN 100	Yes
f0/15	PVLAN 200 Community	PVLAN 300 Isolated	PVLAN 200	Yes
f0/15	PVLAN 200 Community	PVLAN 300 Isolated	PVLAN 300	No

According to the PVLAN configuration guide of the C3560 switch [25], the VLAN filtering of the SPAN port is not supported. Therefore, it is impossible to test VLAN filtering options.

Monitoring the isolated PVLAN attack revealed that both SPAN and VSPAN options are reliable.

The VSPAN configuration which monitored the target PVLAN 300 failed to capture the malicious traffic sent from community PVLAN 200 to isolated PVLAN 300.

The situation is similar to the double-encapsulated VLAN hopping attack which was monitored over the VLAN filtering option. In both cases, monitoring the target VLAN failed to identify the attacker if the attacker sent frames from the source to target VLAN. SPAN debugging failed to provide error messages and switch treated the sent frames as usual traffic.

Although the VSPAN is monitoring the target VLAN, if the ingress and egress interfaces belong to different VLANs, the VSPAN failed to catch the frame passed to the egress interface on the target VLAN.

8.4 Voice VLAN (VoIP) hopping attack

As described in chapter 2.4, working VoIP functionality requires OSI layer 3 device like a router to provide Telephony Services to the switch. Two Switch ports f0/10 and f0/11 configured to support VoIP functionality. Router interface f0/0 configured to provide Telephony Services to the switch.

Table 15: Voice VLAN testing topology.

Device	Interface	IP Address	Subnet Mask (CIDR)	Default Gateway	Switch Port	VLANs
Attacker	NIC	192.168.15.10	/24	192.168.15.1	f0/10	VLAN 10 VLAN 15
Target	NIC	192.168.15.6	/24	192.168.15.1	f0/11	VLAN 10 VLAN 15

VoIP Hopper which is described in chapter 7.3.3 was used for the attack execution. VoIP Hopper automatically discovered the voice VLAN ID from the CDP packet sent by the switch and created the sub-interface for the data VLAN 10 on the operating system.

8.4.1 SNMP and Syslog results

An SNMP and Syslog are configured on the switch S1 and the router R1 to detect the possible reactions. The attack procedure requires plugging off the cable from IP phone and using it on the attacking system.

Therefore only the usual link down and link up messages appeared on the switch logs without any other indication. The SNMP provided the corresponding link down and up messages.

Router failed to provide any reaction to the VLAN hopping event. The keepalive timers of the attacker IP have not expired during the VLAN hopping event.

IP phone creates a trunk link between itself and switch. Voice VLAN hopping is exploiting the trunk link between IP phone and switch which is already in place.

Voice VLAN ID is used by the IP phone to tag its packets and data packets remain untagged. The trunk link allows both data and voice VLANs to become accessible if the attacker mimic the behaviour of the IP phone and creates appropriate target VLAN linked sub-interfaces on the OS.

8.4.2 Compare packet capture methods

VSPAN testing exposed the similar behaviour of the VSPAN logic as discovered by testing Switch Spoofing attacks. If the attacker hopped from the source to the target and

the communication link has been established over the sub-interface which is linked to the target VLAN, then the VSPAN which is monitoring the source VLAN was not able to notice the attacker.

Table 16: SPAN and VSPAN results of the Voice VLAN hopping attacks.

Source Switch Port	**Source VLAN**	**Target VLAN**	**SPAN Source Configurations**	**Attacker Identified**
f0/10	VLAN 15	VLAN 10	f0/10	Yes
f0/10	VLAN 15	VLAN 10	VLAN 10	Yes
f0/10	VLAN 15	VLAN 10	VLAN 15	No
f0/10	VLAN 15	VLAN 10	VLAN 10, VLAN 15	Yes

For example, VSPAN was monitoring VLAN 15 which is also the source of the attacker. If the attacker creates the sub-interface which is linked to the target VLAN 10, then the hopping remained invisible to VSPAN.

VSPAN requires that both source and target VLANs must be monitored to detect any hopping event.

Table 17: VLAN filtering results of Voice VLAN hopping attacks.

Source Switch Port	**Source VLAN**	**Target VLAN**	**Allowed VLANs**	**Attacker Identified**
f0/10	VLAN 15	VLAN 10	VLAN 10	No
f0/10	VLAN 15	VLAN 10	VLAN 15	No
f0/10	VLAN 15	VLAN 10	VLAN 10, VLAN 15	No

VLAN filtering of the trunk link failed to detect any hopping events. The situation is similar to the Switch Spoofing attack where VLAN filtering failed. Both attack types are exploiting trunk links. Testing exposed that SPAN could not handle VLAN filtering of the trunk link if it is under VLAN hopping attack.

9 Summary

VLAN hopping enables to obtain unauthorised access to another VLAN at Ethernet switch level by exploiting various vulnerabilities. VLAN hopping causes a situation where ingress and egress interfaces of the switch belong to different VLANs.

Much knowledge is available on how to harden the switches against various OSI layer 2 attacks, including VLAN hopping attacks. Enough information is available on how to use various tools to execute different VLAN hopping attacks. Almost no information is available on the detection side of the VLAN hopping attacks.

The objective of the thesis was to research how the various monitoring options of the switch can be used for the detection of VLAN hopping attacks?

The hypothesis of the thesis specified that different types of VLAN hopping attacks which are exploited internally of the switch left behind artefacts, patterns or attributes which can be detected by using available monitoring options of the switch.

Typical Catalyst switch has three monitoring options available: a Syslog, an SNMP and Port mirroring (SPAN). The port mirroring itself has four different options for the monitoring:

- Physical ports;
- Trunk ports;
- VLAN based local SPAN (VSPAN);
- VLAN filtering of the trunk ports.

Common port mirroring of each physical port of the switch is not scalable because it is difficult to mirror many switch ports simultaneously or to deploy many sniffers into the network.

The local SPAN has several limitations and options for the data acquisition. Also, particular configuration parameters are required to monitor OSI layer 2 protocols to

preserve source 802.1Q encapsulations. The Ethernet packets might be dropped by the switch in various circumstances due to switch congestion or destination port oversubscription. Switches may also drop the corrupted Ethernet packets.

The first research objective of the thesis questioned whether the VLAN specific VSPAN or VLAN filtering options are more suitable and scalable choices for the monitoring because the particular VLANs or trunk links are mirrored directly by these options which should be more suitable to detect various VLAN hopping events?

If VLANs are monitored as VSPAN sources, then all the traffic of active ports of the monitored VLANs are included and forwarded to the SPAN destination port. If VLAN filtering is applied to source trunk ports, then all the allowed and active VLANs on the trunk link are forwarded to the SPAN destination port.

Mirroring only the physical ports enables to receive the ingress traffic, but VSPAN or VLAN filtering allows the monitoring at the VLAN level. Because multiple VLANs can be monitored at the same time, this should be more suitable to observe VLAN hopping attacks.

The second research objective of the thesis was to identify how the Syslog and SNMP are reacting to the different VLAN hopping attacks, what kind of SNMP trap messages and Syslog messages are displayed during VLAN hopping events?

The research required repetitive executions of different VLAN hopping attacks. Consequently, a research model was necessary to determine the whole research process. After examining various models, a Generic Process Model for network forensics appeared most reliable. This model was adapted and applied to the research process.

Based on these objectives the multiple VLAN hopping attacks were simulated to study the switch reactions and available monitoring options to detect the VLAN hopping events.

The study involved four known VLAN hopping attacks. There are similarities and differences; here is the summary list of most important characteristics of each attack type.

Table 18: A comparison of different VLAN hopping attacks.

Attributes	**Switch Spoofing attack**	**Voice VLAN hopping attack**	**Private VLAN hopping attack**	**Double Encapsulated 802.1Q VLAN hopping attack**
Main vulnerability	DTP protocol	VoIP architecture	PVLAN architecture	IEEE 802.1Q
VLAN hopping mechanism	Malicious trunk link exploitation	Malicious trunk link exploitation	Malicious frames	Malicious frames
Direction	Bidirectional	Bidirectional	Unidirectional	Unidirectional
Scope	All VLANs which allowed over the trunk link	Voice and Data VLANs	All isolated and community PVLANs inside the same PVLAN domain	All available hosts in the target VLANs
Accessibility	Access to the switch port	Access to the IP phone	Requires detailed information about the target	Requires detailed information about the target
Mitigation	Configure access ports, disable DTP and unused ports	Apply hardening to IP phone, CME; use firewall to separate voice and data VLANs; enable Port Security, 802.1X authentication, IPSG, DAI and DHCP snooping	Configure ACL on the router interface or use VLAN based ACL	Configure access ports; isolate default native VLAN 1; disable native VLAN from the trunk link; ensure VLAN tagging on the trunk link

As a result of testing, the Double Encapsulated 802.1Q VLAN hopping attack is perfect in theory but nearly failed at the laboratory environment. This attack tested on Catalyst 2950, 2960 and 3560 switches. Only the C2950 switch processed the double encapsulated frames. Two other switches dropped the frames at ingress interface level. The attack worked only with default configurations and in insufficient circumstances. This attack also required two linked switches to work. As a result, this VLAN hopping attack type is a moderately theoretical threat than an actual threat. However, this attack type cannot be entirely excluded because it might work in other environments or equipment if the switch can process the double encapsulated frames.

Testing revealed that Syslog and SNMP failed to provide adequate reaction to VLAN hopping events. The Syslog agent was configured to operate at the debug level 7, and all the SNMP traps enabled on the switch.

Only the Switch Spoofing attack produced useful SNMP trap message. Other reactions cover common link down and link up messages without any additional log information.

Table 19: An SNMP and Syslog reactions to VLAN hopping attacks.

Attacks	**Switch S1**		**Router R1**		**Switch S2**	
	SNMP	**Syslog**	**SNMP**	**Syslog**	**SNMP**	**Syslog**
Switch Spoofing attack	Yes	reaction	n/a	n/a	n/a	n/a
Double Encapsulated 802.1Q VLAN hopping attack	No	No	n/a	n/a	No	No
Private VLAN hopping attack	No	No	No	No	n/a	n/a
Voice VLAN hopping attack	reaction	reaction	No	No	n/a	n/a

* n/a indicates not applicable.

In summary, the Syslog and SNMP are not suitable options for detecting the VLAN hopping attacks.

The Switch Spoofing and Voice VLAN attacks are exploiting the trunk links. In case of Voice VLAN hopping attack, the exploitable trunk link is already established and available, if exploited, no messages or notifications are given. Switch Spoofing attack reacted if the trunk link established via DTP exploitation which gave notifications during the new trunk link activation process.

PVLAN and Double Encapsulated VLAN hopping attacks are using malicious Ethernet frames for the exploitation which appeared as general traffic for the switches.

According to the research objectives, it is expected that Ethernet switch might provide some system log messages or SNMP trap messages, but testing revealed that these switch monitoring options are unusable and unreliable for the detection. The only exception applies to the Switch Spoofing attack.

The second objective was to research how the different available port mirroring options of the switch are reacting to the various VLAN hopping events? Remind that VLAN

hopping causes a situation where ingress and egress interfaces of the switch belong to different VLANs.

It is expected that VSPAN or VLAN filtering of the trunk links are more suitable monitoring options because these options are operating at VLAN level which should be more suitable options to detect VLAN hopping from one VLAN to another.

Testing exposed controversial results regarding the different port mirroring options. The results are summarised and presented in the table below.

Table 20: A comparison of different SPAN options to detect VLAN hopping attacks.

	Physical source port	**Physical trunk port**	**VSPAN**	**VLAN filtering**
Switch Spoofing attack	Always	Failed	Mirroring target VLAN caught the hopping, not source VLAN	Failed
Double Encapsulated 802.1Q VLAN hopping attack	Always	Always	n/a	Filtering source VLAN caught the hopping, not target VLAN
Private VLAN hopping attack – inside isolated PVLANS	Always	n/a	Always	n/a
Private VLAN hopping attack – Between the community and isolated PVLANs	Always	n/a	Mirroring primary VLAN or source VLAN caught the hopping, not target VLAN	n/a
Voice VLAN hopping attack	Always	Failed	Mirroring target VLAN caught the hopping, not source VLAN	Failed

The PVLAN configuration guide of the Catalyst 3560 switch refers that the PVLAN does not support the VLAN filtering of the SPAN port and the feature is not available (n/a) [25]. The Catalyst 2950 switch does not support VSPAN option.

Trunk links which are carrying all allowed tagged and untagged VLAN traffic is expected to provide a good option for the VLAN hopping detection.

VSPAN monitoring performed differently in trunk links compared to sending malicious frames over the trunk link. Exploiting the trunk links allowed VSPAN to identify the hopping if the target VLAN was monitored. VSPAN identified sent malicious frames if source VLAN was monitored. In both cases, the hopping detected if both source and target VLANs were monitored.

Unfortunately, the testing revealed that in many cases this VLAN filtering option is either not available, failed to operate or was unreliable in terms of monitoring the VLAN hopping attacks.

VLAN filtering of the trunk link failed on both Switch Spoofing and Voice VLAN attacks while the attacker exploited the trunk link at the same time when the trunk link was configured to monitor VLANs over the trunk link. Unfortunately, the switch failed to provide debug error messages.

Another option, VSPAN performed better but provided unreliable results if only one VLAN monitored as a source VLAN. The VLAN hopping occurs from VLAN to another, which means that the attacker can hop from the source subnet to the different target subnet.

Testing exposed that VSPAN behaves similarly on Switch Spoofing and Voice VLAN attacks because both these attacks are exploiting the trunk links. For example, the VSPAN is monitoring VLAN X which is also the source VLAN of the attacker. If the attacker creates the sub-interface which linked to the target VLAN Y, then the hopping from VLAN X to Y was entirely invisible to the VSPAN. VLAN hopping event enabled traffic to bypass the source VLAN monitoring because the new channel established between sub-interface which is participating in the target VLAN and the host in target VLAN.

PVLAN hopping attacks between the community and isolated PVLANs are similar to Double Encapsulated VLAN hopping attack because both attacks are sending malicious frames from one VLAN to another. VLAN filtering testing revealed a similar behaviour. In both cases, monitoring the target VLAN failed to identify the attacker if the attacker

sent frames from the source to target VLAN. Although the VSPAN is monitoring the target VLAN, if the ingress and egress interfaces belong to different VLANs, the SPAN failed to catch the frame passed to the egress interface on the target VLAN. Switch treated the sent frames as usual traffic and debugging failed to provide error messages.

In summary, the traditional source port mirroring method worked always, but this method requires monitoring every port which has scalability issues on big networks. The VSPAN is an alternative option if both source and target VLANs are monitored, otherwise it is not suitable. VLAN filtering of the trunk links failed or is unreliable and is not a suitable option. A Syslog is unusable to detect any attack and SNMP is only useful to detect Switch Spoofing VLAN hopping attacks.

References

[1] E. Vyncke and C. Paggen, *LAN Switch Security: What Hackers Know About Your Switches*. Indianapolis, IN: Cisco Press, 2008.

[2] Crowd Research Partners, "Insider Threat Report," 2018 [Online]. Available: https://www.ca.com/content/dam/ca/us/files/ebook/insider-threat-report.pdf. [Accessed: 09-Apr-2018]

[3] T. Szabó, "Network security problems on layer 2," *Hadmérnök*, vol. 8, no. 1, 2013.

[4] B. Cusack and R. Lutui, "Innovating additional Layer 2 security requirements for a protected stack," presented at the Australian Information Security Management Conference, Auckland University of Technology, 2015, pp. 81–86.

[5] H. Altunbasak, S. Krasser, H. L. Owen, J. Grimminger, H.-P. Huth, and J. Sokol, "Securing layer 2 in local area networks," presented at the International Conference on Networking, 2005, pp. 699–706.

[6] H. Altunbasak, S. Krasser, H. Owen, J. Sokol, and J. Grimminger, "Addressing the weak link between layer 2 and layer 3 in the Internet architecture," presented at the 29th Annual IEEE International Conference on Local Computer Networks, Tampa, FL, USA, USA, 2004, pp. 417–418.

[7] D. Liu, *Cisco Router and Switch Forensics: Investigating and Analyzing Malicious Network Activity*. Burlington, MA: Syngress, 2009.

[8] W. Odom, *CCENT/CCNA ICND1 100-105 Official Cert Guide*, 1st. edition. Indianapolis, IN: Cisco Press, 2016.

[9] G. Leischner and C. Tews, "Security through VLAN segmentation: Isolating and securing critical assets without loss of usability," presented at the 9th Annual Western Power Delivery and Automation Conference, Spokane, Washington, 2007, pp. 1–7.

[10] J. Akram, N. Akram, S. Mamoon, S. Ali, and N. Naseer, "Future and Techniques of Implementing Security in VLAN," *Journal of Network Communications and Emerging Technologies (JNCET)*, vol. 7, no. 5, 2017.

[11] V. Umasuthan, "Protecting the Communications Network at Layer 2," presented at the 2016 IEEE/PES Transmission and Distribution Conference and Exposition (T & D), Dallas, TX, USA, 2016, pp. 1–5.

[12] S. A. Rouiller, "Virtual LAN Security: weaknesses and countermeasures," *SANS Institute InfoSec Reading Room*, 2003 [Online]. Available: https://www.sans.org/reading-room/whitepapers/networkdevs/virtual-lan-security-weaknesses-countermeasures-1090

[13] R. L. Bull, J. N. Matthews, and K. A. Trumbull, "VLAN hopping, ARP poisoning and Man-In-The-Middle Attacks in Virtualized Environments," presented at the DEF CON 24, At Las Vegas, NV, 2016, p. 9 [Online]. Available: https://www.researchgate.net/publication/307509482_VLAN_hopping_ARP_poisoning_and_ Man-In-The-Middle_Attacks_in_Virtualized_Environments

[14] "Ethernet switch network market share worldwide 2011-2017 | Statistic," *Statista*. [Online]. Available: https://www.statista.com/statistics/235289/global-ethernet-switch-revenue-market-share-by-vendors/. [Accessed: 11-Apr-2018]

[15]"System Message Logging," *Cisco*. [Online]. Available: https://www.cisco.com/c/en/us/td/docs/routers/access/wireless/software/guide/SysMsgLogging.html. [Accessed: 02-Apr-2018]

[16]"SNMP Configuration Guide, Cisco IOS Release 12.4T - Configuring SNMP Support [Support]," *Cisco*. [Online]. Available: https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/snmp/configuration/12-4t/snmp-12-4t-book/nm-snmp-cfg-snmp-support.html. [Accessed: 02-Apr-2018]

[17]"Catalyst 2960, 2960-S, and 2960-P Switch Software Configuration Guide, Cisco IOS Release 15.0(2)EZ - Configuring SPAN and RSPAN," *Cisco*. [Online]. Available: https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst2960/software/release/15-0_2_ez/configuration/guide/scg_2960/swspan.html. [Accessed: 18-Apr-2018]

[18]"Using the Cisco Span Port for San Analysis," *Cisco*. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/data-center-virtualization/san-consolidation-solution/net_implementation_white_paper0900aecd802cbe92.html. [Accessed: 08-Apr-2018]

[19]E. S. Pilli, R. C. Joshi, and R. Niyogi, "Network forensic frameworks: Survey and research challenges," *Digital Investigation*, vol. 7, no. 1, pp. 14–27, Oct. 2010 [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1742287610000113. [Accessed: 03-Apr-2018]

[20]R. Froom and E. Frahim, *Implementing Cisco IP Switched Networks (SWITCH) Foundation Learning Guide*. Indianapoils, IN: Cisco Press, 2015.

[21]"How to Secure Cisco Routers and Switches." [Online]. Available: https://www.globalknowledge.com/us-en/content/articles/how-to-secure-cisco-routers-and-switches/. [Accessed: 09-Apr-2018]

[22]O. Santos, *CCNA Security 210-260 Official Cert Guide*, 1st edition. Indianapolis, IN: Cisco Press, 2015.

[23]P. Browning, F. Tafa, D. Gheorghe, and D. Barinic, *Cisco CCNA in 60 Days*. Milton Keynes, UK: Reality Press Ltd., 2014.

[24]S. HomChaudhuri and M. Foschiano, "Cisco Systems' Private VLANs: Scalable Security in a Multi-Client Environment." [Online]. Available: https://tools.ietf.org/html/rfc5517. [Accessed: 20-Apr-2018]

[25]"Catalyst 3560 Software Configuration Guide, Release 12.2(52)SE - Configuring Private VLANs," *Cisco*. [Online]. Available: https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst3560/software/release/12-2_52_se/configuration/guide/3560scg/swpvlan.html. [Accessed: 19-Apr-2018]

[26]"Cisco Nexus 5000 Series NX-OS Software Configuration Guide - Configuring Private VLANs," *Cisco*. [Online]. Available: https://www.cisco.com/c/en/us/td/docs/switches/datacenter/nexus5000/sw/configuration/guide/cli/CLIConfigurationGuide/PrivateVLANs.html. [Accessed: 11-Apr-2018]

[27]S. K. Sonkar, R. Singh, R. Chauhan, and A. P. Singh, "A Review Paper: Security on Voice over Internet Protocol from Spoofing attacks," vol. 1, no. 3, p. 8.

[28]J. Cioara and M. Valentine, *CCNA Voice 640-461 Official Cert Guide*, 2nd ed. Indianapolis, IN : London: Cisco Press ; Pearson Education [distributor], 2012.

[29]O. Santos, *Network Security with NetFlow and IPFIX: Big Data Analytics for Information Security*. Indianapolis: Cisco Press, 2016.

[30] S. Davidoff and J. Ham, *Network Forensics: Tracking Hackers Through Cyberspace*. Upper Saddle River, NJ: Prentice Hall, 2012.

[31] C. Sanders and J. Smith, *Applied Network Security Monitoring: Collection, Detection, and Analysis*. Boston: Syngress, Elsevier Inc., 2014.

[32] O. Santos, *CCNA Cyber Ops SECOPS 210-255 Official Cert Guide*, 3rd edition. Indianapolis, IN: Cisco Press, 2017.

[33] W. Odom, *CCNA Routing and Switching ICND2 200-105 Official Cert Guide*, 1st. edition. Indianapolis, IN: Cisco Press, 2017.

[34] "Simple Network Management Protocol." [Online]. Available: http://telescript.denayer.wenk.be/~hcr/cn/idoceo/udp_snmp.html. [Accessed: 08-Apr-2018]

[35] "Catalyst Switched Port Analyzer (SPAN) Configuration Example," *Cisco*. [Online]. Available: https://www.cisco.com/c/en/us/support/docs/switches/catalyst-6500-series-switches/10570-41.html. [Accessed: 02-Apr-2018]

[36] M. A. Qadeer, A. Iqbal, M. Zahid, and M. R. Siddiqui, "Network Traffic Analysis and Intrusion Detection Using Packet Sniffer," presented at the 2010 Second International Conference on Communication Software and Networks, 2010, pp. 313–317.

[37] "CVE Search Results," *Common Vulnerabilities and Exposures*. [Online]. Available: https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=VLAN. [Accessed: 03-May-2018]

[38] Y. Bhaiji, "Understanding, Preventing, and Defending Against Layer 2 Attacks," Cisco, 2007 [Online]. Available: https://www.cisco.com/c/dam/global/en_ae/assets/exposaudi2009/assets/docs/layer2-attacks-and-mitigation-t.pdf

[39] "VLAN-Based Network Attacks : Switching Security: LAN Switching First-Step," *eTutorials.org*. [Online]. Available: http://etutorials.org/Networking/Lan+switching+first-step/Chapter+9.+Switching+Security/VLAN-Based+Network+Attacks/. [Accessed: 11-Apr-2018]

[40] "VLAN Hacking," *InfoSec Resources*, 08-Dec-2011. [Online]. Available: http://resources.infosecinstitute.com/vlan-hacking/. [Accessed: 11-Apr-2018]

[41] A. M. Jingi and M. Muhammad, "VoIP Security: Common Attacks and their Countermeasures," *International Journal of Computer Science and Information Security*, vol. 15, no. 3, p. 421, 2017.

[42] S. McGann and D. C. Sicker, "An Analysis of Security Threats and Tools in SIP-Based VoIP Systems," presented at the Second VoIP security workshop, 2005, p. 8.

[43] J. Ostrom and J. Kindervag, "VoIP Hopping: A Method of Testing VoIP security or Voice VLANs," *Symantec Connect Community*. [Online]. Available: https://www.symantec.com/connect/articles/voip-hopping-method-testing-voip-security-or-voice-vlans. [Accessed: 11-Apr-2018]

[44] "Cisco Unified Communications Manager Security Guide, Release 8.6(1) - Phone Hardening," *Cisco*. [Online]. Available: https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/security/8_6_1/secugd/sec-861-cm/secu_ph.html. [Accessed: 06-May-2018]

[45] D. Persky and J. Niem, "VoIP Security Vulnerabilities," SANS Institute InfoSec Reading Room, 2007 [Online]. Available: https://www.sans.org/reading-room/whitepapers/voip/voip-security-vulnerabilities-2036

[46] Jianqiang, Xin, “Security Issues and Countermeasure for VoIP,” SANS Institute InfoSec Reading Room, 2007 [Online]. Available: https://www.sans.org/reading-room/whitepapers/voip/security-issues-countermeasure-voip-1701

[47] R. C. Joshi, *Fundamentals of Network Forensics: A Research Perspective*. New York, NY: Springer Berlin Heidelberg, 2016.

[48] M. Rasmi, A. Jantan, and H. Al-Mimi, “A New Approach for Resolving Cyber Crime in Network Forensics Based on Generic Process Model,” presented at the 6th International Conference on Information Technology (ICIT 2013), 2013.

[49] “Yersinia.” [Online]. Available: http://www.yersinia.net/. [Accessed: 19-Apr-2018]

[50] “Yersinia,” *Kali Tools*. [Online]. Available: https://tools.kali.org/vulnerability-analysis/yersinia. [Accessed: 19-Apr-2018]

[51] “packETH.” [Online]. Available: http://packeth.sourceforge.net/packeth/Home.html. [Accessed: 19-Apr-2018]

[52] “VoIP Hopper - Jumping from one VLAN to the next!” [Online]. Available: http://voiphopper.sourceforge.net/. [Accessed: 19-Apr-2018]

[53] “VoIPHopper,” *Kali Tools*. [Online]. Available: https://tools.kali.org/sniffingspoofing/voiphopper. [Accessed: 19-Apr-2018]

[54] “Cisco IOS MIB Locator,” *Cisco*. [Online]. Available: http://mibs.cloudapps.cisco.com/ITDIT/MIBS/servlet/index. [Accessed: 19-Apr-2018]

10 Appendix 1 – Switch configurations

Switch configurations for the Switch Spoofing and Double Encapsulation VLAN hopping attacks.

```
hostname S1
no ip domain-lookup

line con 0
 exec-timeout 0 0
 logging synchronous
 exit

Vlan 10
 name Data10
 exit
Vlan 15
 name Voice15
 exit
vlan 20
 name Data20
 exit
Vlan 50
 name Native
 exit
Vlan 99
 name Management
 exit
Vlan 80
 name SPAN
 exit

interface vlan 99
 ip address 192.168.99.10 255.255.255.0
 no shut
 exit

int f0/23
 switchport access vlan 99
 Description SYLOG, SNMP
 no shut
 exit

int f0/24
 Description SPAN
```

```
 switchport access vlan 80
 no shut
 exit

snmp-server community public ro
snmp-server host 192.168.99.30 version 2c public
snmp-server enable traps

logging on
logging console 7
service timestamps log datetime msec
service sequence-numbers
logging host 192.168.99.30
logging trap 7
```

Switch configurations for the PVLAN attacks:

```
vtp mode transparent

vlan 200
 name Communitty
 private-vlan community
 exit

vlan 300
 name Isolated
 private-vlan isolated
 exit

vlan 100
 name primary
 private-vlan primary
 private-vlan association 200,300
 exit

int f0/2
 switchport mode private-vlan promiscuous
 switchport private-vlan mapping 100 200,300
 no shut
 exit

int range f0/15-16
 switchport mode private-vlan host
 switchport private-vlan host-association 100 200
 no shut
 exit

int range f0/17-18
 switchport mode private-vlan host
 switchport private-vlan host-association 100 300
 no shut
```

```
 exit
```

Switch configurations for Voice VLAN attacks:

```
int f0/1
 switchport trunk encapsulation dot1q
 switchport mode trunk
 switchport trunk native vlan 50
 no shut
 exit

int range f0/10-11
 switchport mode access
 switchport voice vlan 10
 switchport access vlan 15
 exit
```

11 Appendix 2 – Router configurations

```
hostname R1
no ip domain-lookup

logging on
logging console 7

line con 0
 exec-timeout 0 0
 logging synchronous
 exit

int f0/0.10
 encapsulation dot1q 10
 ip address 192.168.10.1 255.255.255.0
 exit
int f0/0.15
 encapsulation dot1q 15
 ip address 192.168.15.1 255.255.255.0
 exit
int f0/0.50
 encapsulation dot1q 50 native
 exit
int f0/0
 no shut
 exit

int f0/1
 ip address 10.0.0.1 255.255.255.0
 no shut
 exit

ip dhcp excluded-address 192.168.10.1 192.168.10.5
ip dhcp excluded-address 192.168.15.1 192.168.15.5

ip dhcp pool DATA10
 network 192.168.10.0 255.255.255.0
 default-router 192.168.10.1
 exit

ip dhcp pool VOICE15
 network 192.168.15.0 255.255.255.0
 default-router 192.168.15.1
 option 150 ip 192.168.15.1
 exit

telephony-service
 max-dn 10
```

```
 max-ephones 10
 ip source-address 192.168.15.1 port 2000
 auto assign 1 to 9
 create cnf-files
 exit

ephone-dn 1 dual-line
 number 1010
 name Phone-1
 label Phone-1
 description Phone-1
 exit

ephone-dn 2 dual-line
 number 1020
 name Phone-2
 label Phone-2
 description Phone-2
 exit

ephone 1
 button 1:1
 type CIPC
 reset
 exit

ephone 2
 button 1:2
 type CIPC
 reset
 exit
```

Router configuration to test an SNMP and Syslog:

```
int f0/1
 ip address 192.168.99.15 255.255.255.0
 no shut
 exit

int f0/0
 ip address 192.168.99.15 255.255.255.0
 no shut
 exit
snmp-server community public ro
snmp-server host 192.168.99.30 version 2c public
snmp-server enable traps
logging on
service timestamps log datetime msec
service sequence-numbers
logging host 192.168.99.30
logging trap 7
```

12 Appendix 3 – SPAN configurations

Various SPAN example configuration parameters. The parameters altered according to various testing scenarios.

SPAN physical port configurations:

```
monitor session 1 source interface f0/3 both
```

VLAN based local SPAN (VSPAN) configurations:

```
monitor session 1 source vlan 1 , 10 , 20 both
```

VLAN filtering configurations:

```
monitor session 1 source interface f0/7 both
monitor session 1 filter vlan 1 , 20
```

If trunk link is monitored directly:

```
switchport trunk allowed vlan 1
```

Destination port configurations:

```
monitor session 1 destination interface Fa0/24 encapsulation replicate

monitor session 1 destination interface Fa0/24 encapsulation dot1q

monitor session 1 destination interface Fa0/24 encapsulation replicate
ingress dot1q vlan 1
```

13 Appendix 4 – Debugging and troubleshooting commands

Debugging and troubleshooting commands to extract volatile and non-volatile information from the switch. The commands used differently according to various testing scenarios.

```
debug monitor all
debug mac-notification
debug ip address
debug ip icmp
debug ip arp track
debug ip socket
debug ethernet interface
debug interface vlan 1
```

Commands to clear volatile information:

```
clear mac address-table dynamic
clear mac address-table notification
clear logging
clear cef table ipv4
clear counters
clear interface fastEthernet 0/3
```

Troubleshooting commands:

```
show mac address-table dynamic
show ip cef switching statistics
show interface f0/3
show ip interface f0/3
show interface f0/3 controller
show buffers input-interface f0/3
show buffers input-interface vlan 1
show platform port-asic stats drop
show platform port-asic span 1
show platform port-asic mac-info f0/3
Show ip cef switching statistics
show monitor detail
show monitor capture buffer all parameters
show monitor capture point all
show monitor event-trace all-traces parameters
```