CRYPTOGRAPIC SECURITY ANALYSIS OF IPV6 ON VPN USING A BOTNETS

A Thesis

SUBMITTED TO THE

TILAK MAHARASHTRA VIDYAPEETH,
PUNE

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

In Computer Science

Under the Board of Modern Sciences & Professional Skills

BY

Mrs. Rama Sanjay Bansode
(25315008505)

UNDER THE GUIDANCE OF

Dr. ANUP GIRDHAR

DEPARTMENT OF COMPUTER SCIENCE

JULY 2022

CERTIFICATE BY THE SCHOLAR

This is to certify that the thesis titled**, "CRYPTOGRAPHIC SECURITY ANALYSIS OF IPV6 ON VPN USING A BOTNETS"**
submitted by Mrs. Rama Sanjay Bansode, under the supervision of Dr. Anup Girdhar, P.R.N. Number 25315008505 for the award of Ph.D. degree of Computer science from 'Tilak Maharashtra Vidyapeeth' University embodies my original work and has not formed the basis for the award of any degree, diploma, fellowship, titles in this or any other university or any other similar institutions of higher learning.

Place:

Date:

Signature

Mrs. Rama Sanjay Bansode

Research Scholar

Tilak Maharashtra Vidyapeeth Pune.

Enrollment Number 25315008505

CERTIFICATE BY THE SUPERVISOR

This is to certify that the work incorporated in the thesis **"CRYPTOGRAPHIC SECURITY ANALYSIS OF IPV6 ON VPN USING A BOTNETS"**

submitted by Mrs. Rama Sanjay Bansode is an original piece of research work carried out by the candidate under my supervision. Literary presentation is satisfactory and the thesis is in a form suitable for publication. Work evinces the capacity of the candidate for critical examination and independent judgment. Such materials as has been obtained from other sources have beenduly acknowledged in the thesis.

Place:

Date:

Signature

Dr. Anup Girdhar, Ph.D.

Research Guide, TMV

DECLARATION

I hereby declare that this Ph.D. thesis entitled “Cryptographic Security Analysis of IPv6 on VPN Using a Botnets" was carried out by me for the degree of Doctor of Philosophy in Computer Science under the guidance and supervision of Dr. Anup Girdhar, Tilak Maharashtra Vidyapeeth, Pune, India. The interpretations put forth are based on my own experimentation and understanding. The books, articles and websites, which I have made use of are acknowledged at the respective place in the text. All content is plagiarism free. All the information gained is from genuine sources and the results are obtained following the detailed procedure. For the present thesis, which I am submitting to the University, no degree or diploma or distinction has been conferred on me before, either in this or in any other University.

Place:

Date:

Signature

Mrs. Rama Sanjay Bansode

PRN NO. 25315008505

Research Scholar

Tilak Maharashtra Vidyapeeth Pune.

ACKNOWLEDGEMENTS

I wish to sincerely thank all those who have contributed in one way or another to this study. Words can only inadequately express my deep gratitude to my guide, **Dr. Anup Girdhar,** for his fruitful comments and insightful suggestions have been a crucial formative influence on the present study. He has supported me in every possible way since the beginning of my research. Without his guidance and encouragement, my research would have never come out in the present form. A special thanks **Mrs. Asmita Namjoshi**, Head of TMV, Computer Science Department for their support and facilities provided. I wish to express my sincere gratitude to **Dr. Sunanda Yadav,** the Director of the place of research, for her expert guidance and invaluable suggestions. I am deeply grateful to my husband **Mr. Sanjay Bansode** who constantly supported and inspired me and helped me throughout this journey. I would also like to thank my daughters **Akshadha Bansode and Swara Bansode** who constantly co-operated with me in every possible way. I would also like to acknowledge **my parents, my brother's and sister's** , for their constant moral support . I am also indebted to **my in-laws** for their continuous trust, encouragement and faith in me through the ups and downs of my research work . I am particularly grateful my friend **Mrs. Swati Ghule**. I would like to thank her, she was always willing to help and give best suggestions.

I would also like to the Management, **Dr. G. R. Ekbote Chairman,**

P.E.Society, Pune for their valuable support. I thank **Dr. (Mrs.) K. R. Joshi, Principal** and **MCA Department HOD, teaching and non teaching staff** of my Institute, P. E. Society's Modern College of Engineering, Pune for supporting me through out my research work. Last, but not the least I would like to extend my gratitude to all those individuals who have helped me directly or indirectly in the completion of this research activity.

Mrs. Rama Sanjay Bansode
Research Scholar, TMV.

Tilak Maharashtra Vidyapeeth, Pune

Undertaking

1. I _____ Mrs. Rama Sanjay Bansode,_____ have registered my

   name for the Ph.D. Course in Computer Science _____ in the year 2015-16 .

2. The undertaken research is entitled as : " CRYPTOGRAPHIC SECURITY ANALYSIS OF IPV6 ON VPN USING A BOTNETS"

3. I have gone through extensive review of literature of the related published / unpublished research works and the use of such references made have been acknowledged in my thesis.
4. The title and the content of research is original.
5. I understand that, in case of any complaint especially plagiarism, regarding my Ph.D. research from any party, I have to go through the enquiry procedure as decided by the Vidyapeeth at any point of time.
6. I understand that, if my Ph.D. thesis (or part of it) is found duplicate at any point of time, my research degree will be withdrawn and in such circumstances, I will be solely responsible and liable for any consequences arises thereby. I will not hold the TMV, Pune responsible and liable in any case.

I have signed the above undertaking after reading carefully and knowing all the aspects there in.

Signature : _____

Address : F 602 WATERS EDGE VISHALNAGAR PUNE

Ph.No. : 9921325936 e-mail : rama.bansode@yahoo.com

Date : _____ Place : Pune

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

IoT	: Internet of Things
IPv6	: Internet Protocol Version 6
IETF	: Internet Engineering Task Group
SLAAC	: Stateless address autoconfiguration
VPN	: Virtual Private Network
ISP	: Internet Service Provider
DDoS	: Distributed Denial of Service
IDM	: Identity Management
DHCP	: Dynamic Host Control Protocol
SDN	: Software-defined network
ARPANET	: American military computer network
NSFNET	: Nation Science Foundation”
PSN	: Packet network switched
DTE	: Data Terminal Equipment
DCE	: Data Communication Equipment
SVC	: Switch Virtual Circuit
PVC	: Permanent Virtual Circuit
PAD	: Packet Assembler Disassembler
FR	: Frame Relay
FECN	: Forward Explicit Notification for Congestion
BECN	: Backward Explicit Congestion Notice
GRE	: Generic Routing Encapsulation
CVE	: Common Vulnerability Exposure
MAC	: Media Access Control
SNMP	: Simple Network Management Protocol
NDP	: Neighbor Discovery Protocol
NTP	: Network Time Protocol
TLLA	: Target Link-Layer Address
DAD	: duplicate address detection
PKI	: Public Key Infrastructure
SDN	: Software-defined Network.
EDA	: Exploratory Data Analysis
MSDNAA	: Microsoft Developer Network Academic Alliance

TABLE OF CONTENTS

ABSTRACT

Everyone is currently online, connected to the internet, the internet has become our necessities, thanks to its easy access and access to nearly any service and assistance. The Indian government also encourages digitalization and offers a variety of digital services such as e-government and digital payments. On the other hand, illiteracy remains a major problem in India, particularly in rural regions. Adopting and trusting new technologies is difficult. At the same time, new concepts such as the Internet of Things (IoT), high transmission speeds on 4G / 5G, and so on have been launched. On the other hand, for a variety of reasons, hacking attempts have increased. There have been thousands of malware introduced. Digital proof is based on details of device connectivity through IP addresses and other artifacts on the research side. However, because of the increase of worldwide network devices IPv4, IPv4 is the best way of connecting digital devices. With IPv4 configurations, many further limitations have evolved as well. To solve this problem, however, IPv6 remains adaptable based on any organization's networks, devices, and tailor-made requirements. Virtual private networks are one type of network users that connect many of the company's websites and employees worldwide can access their internal resources and become part of their closed network. VPN types can be configured in many ways, according to customized requirements. However, hackers expose and exploit many vulnerabilities on many occasions. The transformation of IPv6 will certainly support the identification of geo-locations and other digital device artifacts and the mitigation and adaptability of existing equipment and the configuration of the infrastructure of the latest risk factors. In this study, the focus of the study is the identification of the risks, challenges, and opportunities of VPN implementation using IPv6 and the protection of the infrastructure against existing malware as botNets.

Benefit of the Study

This research will help the whole world in particular the government, agencies, businesses, network solutions architects, developers of IDPs and policymakers, etc. These are of great help. Without a doubt, this study will explore the hidden challenges of cybersecurity and the latest technologies for implementing VPNs with IPv6. The aim is to propose a security framework that respects confidentiality, completeness, availability, and authenticity. In addition, this study will provide a transparent, cost-effective, and reliable model to prevent the challenges and threats that will arise in the field of network security. This study would certainly protect the critical applications services, data, and digital identity against identified or unspecified malware from financial losses.

Proposed Model

The proposed framework model combines the implementation and analysis of the different phases involved in preparing a secured model. Legitimate users as well as Botnet's are constantly targeting an Organization's Network

Infrastructure, however, organizations have been shown to apply the generic VPN process and to take security less priority due to the lack of technical, financial, and other internal organizations, thus compromising the infrastructure. In the proposed model, the goal is to provide an adequate means of implementing the VPN infrastructure, which is classified into six categories:

1. Authentication and Authorization
2. Application testing and integration
3. Privilege acceleration
4. Log Analysis, Data Collection, and Sanitization
5. Data Analysis and Synchronization of Trained Engine
6. Repository updates and Model Synchronization

Objectives of the Framework Implementation.

Step1: The first stage, as stated in the proposed model, is to ensure that the secure VPN infrastructure is implemented properly. While legitimate users and Botnets hit the VPN, generic VPN implementations lacked security policies, the first step proposed in the model was
end-to-end encryption between remote users and front-end routers.
Step2: Firewall-1 should prevent all irrelevant Internet requests, and for internal authentication, only VPN requests are to be processed and forwarded.
Step3: A user authentication server (UAS) (Authentication Server, 2019) is needed for application
validation for internal authentication. The policy at this stage must determine whether the requirement comes from the trusted device or not, if the system is trusted, the request must be
processed by UAS and the End to End asymmetric encryption applied is decrypted. If the device is not trusted, then the two-stage verification model will be implemented and the user
and manager will be notifiable.
Figure 3.29: Proposed ModelStep4: If everything goes smoothly to complete end-to-end decryption, a TLS token for
specific support needs to be created for UAS servers. The additional token was forwarded for
the token verification to Firewall-2. The concept is based on the customized package and techniques of token passing.
Step5: Now if the token is checked then only the request must be transported to the VPN server
and sent via static or DHCPv6 servers to allocating IPv6.
Step6: Now the public/private IPv6 chain IPv6 address with the combination of MAC address
on the confident device is allowed according to the VPN server requirements.

Step7: Once IPv6 has been allocated to the end-user, they can access services such as Application Servers Stack, Branch Office, LAN cluster, and other web-based services that are
based on the specific user policy. Once the user is approved, in legacy VPN, there is a privilege
to access all network services and ports. However, it has been mitigated in this proposed framework.
Step8: All activities are registered on the centralized log server, and behavior analysis and detection techniques are further processed.
Step9: The framework should be trained with known samples and samples collected.
Step 10: The framework is tested using live wild samples.
Step 11: A limited and future scope analysis of the proposed framework.

In our approach, The entire Framework is based on the combination of various stages to accomplish the expected outcomes. The final Algorithm is completely based on the six major stages of the proposed framework.

1. **The Final Algorithm:**

Let us assume the NGVF is the multifactor set:

NGVF={f(a) U f(b) U f(c) U f(d) U f(e) U f(f)}

Where:

NGVF=Next Gen VPN Framework.

f(a) →Authentication and Authorization Analysis

f(b) →Application Testing and Integration Analysisf(c) →Privilege Acceleration Analysis

f(d) →Log Analysis, Data Collection, and Sanitization

f(e) →Data Analysis and Synchronization of Trained Engine

f(f) →Repository updates and Model Synchronization

2. **Authentication and Authorization Analysis:**

This is the initial section of the proposed framework which works on Authentication and the Authorization of the visitor.

Authentication Module: This module includes the user activities to interact with identity management which identifies the trusted/ legitimate systems and generates the token. It also helps to generate the authenticated permissions to grant access. All the activities are logged and updated in the database.

3. **Application Testing and Integration Analysis**

This section includes the second phase of the framework, where user anonymity, application permissions, and integration are required.

4. **Privilege Acceleration Analysis**

This module is the heart of the framework. It takes cares of the privilege acceleration of the entire system. All the available listed applications, integrated applications, trusted networks and their resources, integrated trusted devices, policies, and permissions are mentioned in this module.

5. **Log Analysis, Data Collection, and Sanitization**

In this phase, it is the process of behavior analysis of the entire framework. It includes the classification and collection of the dataset, data sanitization, purification, and further classification of the activities performed in the framework

6. **Data Analysis and Synchronization of Trained Engine Dataset**

This phase includes the behavior analysis of filtered data/ logs, updation of the dataset, and synchronization with the trained engine of all the activities.

CHAPTER 1. INTRODUCTION

1.1. Introduction to Study

Everyone is currently online, connected to the internet, the internet has become our necessities, thanks to its easy access and access to nearly any service and assistance. The Indian government also encourages digitalization and offers a variety of digital services such as e-government and digital payments. On the other hand, illiteracy remains a major problem in India, particularly in rural regions. Adopting and trusting new technologies is difficult. At the same time, new conceptssuch as the Internet of Things (IoT), high transmission speeds on 4G / 5G, and so on have been launched. On the other hand, for a variety of reasons, hacking attempts have increased. There have been thousands of malwares introduced. Digital proof is based on details of device connectivity through IP addresses and other artifacts on the research side. However, because of the increase of worldwide network devices IPv4, IPv4 is the best way of connecting digital devices. With IPv4 configurations, many further limitations have evolved as well. To solve this problem, however, IPv6 remains adaptable based on any organization's networks, devices, and tailor-made requirements.

Virtual private networks are one type of network users that connect many of the company's websites and employees worldwide can access their internal resources and become part of their closed network. VPN types can be configured in many ways, according to customized requirements. However, hackers expose and exploit many vulnerabilities on many occasions. Thetransformation of IPv6 will certainly support the identification of geo-locations and other digital device artifacts and the mitigation and adaptability of existing equipment and the configuration ofthe infrastructure of the latest risk factors.

In this study, the focus of the study is the identification of the risks, challenges, and opportunitiesof VPN implementation using IPv6 and the protection of the infrastructure against existing malware as botnets.

1.2. Introduction of IPv6

Version 6 (IPv6) (IPv6, 2020) is the latest version of the Internet Protocol (IP), which provides an identification and location system for the Internet traffic on the networks and tunnels. To addressthe long-anticipated problem of IPv4 address exhaustion, IPv6 has been developed by the InternetEngineering Task Group (IETF). The purpose of IPv6 is to substitute IPv4 (IPv4, 2020). IPv6 was later ratified as an Internet Standard on dated 14 July 2017 and was amended as a draught IETF Standard in December 1998.

Internet devices are provided with a unique IP address for identification and definition of location.The rapid growth of the Internet after marketing in the nineties demonstrated the need for far moreaddresses than the IPv4 address space available for connected devices. By 1998, a successor protocol was formalized by the Internet Engineering Task Force (IETF). IPv6 uses an address of 128 bits that enables theoretically allows 2^128 combinations or 340 (trillion)[3] addresses. The actual number is somewhat smaller; as several ranges are for a special purpose reserved or are notcompletely used. The total number of IPv6 addresses by contrast of IPv4 permits 2^32 combinations for a maximum of approximately 4.7 billion addresses that can be obtained by 32- bit addresses. The two protocols are not interoperable and therefore direct communication is not possible among them, which makes it difficult to move to IPv6. Several mechanisms of transitionhave, however, been developed to correct this.

In addition to a larger addressing space, IPv6 offers other technical advantages. It allows hierarchical methods of address allocation, in particular, which allow route aggregation through the Internet and thus limits routing table's expansion. The use of multicast addressing is expanded and simplified and offers further optimization for service delivery. The design of the protocol hastaken into account device mobility, security, and configuration aspects. Eight column segregated groups with four hexadecimal digits represent IPv6 addresses. The full display can be simplified withseveral notation methods; for example —, 2001:0db8:0000:0000:0000:0000:0000:0000:8a2e:0370:7334 is 2001:db8:8a2e:370:7334

IPv6's Address architecture is set in RFC 4291 and allows three different transmission types:unicast, anycast, and multicast.

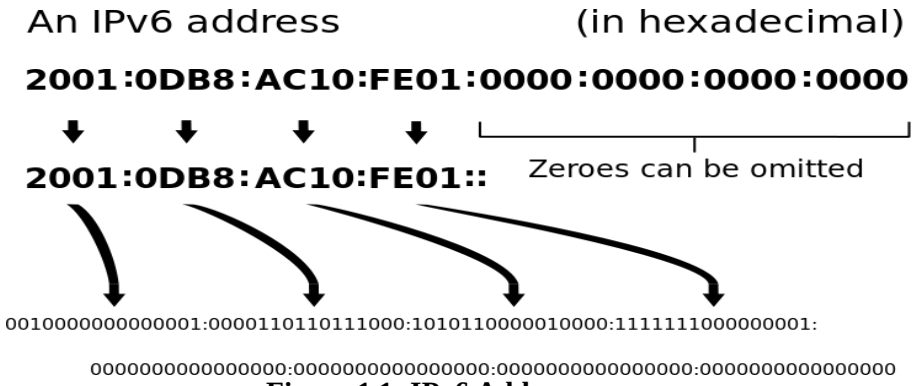


Figure 1.1: IPv6 Address

1.2.1. The main features of IPv6 are as follows:

- Larger address space
- Multicasting
- Stateless address autoconfiguration (SLAAC)
- SLAAC privacy extensions
- IPsec
- Simplified processing by routers
- Mobility
- Extension headers
- Jumbo grams
- Shadow networks
- IPv6 packet fragmentation

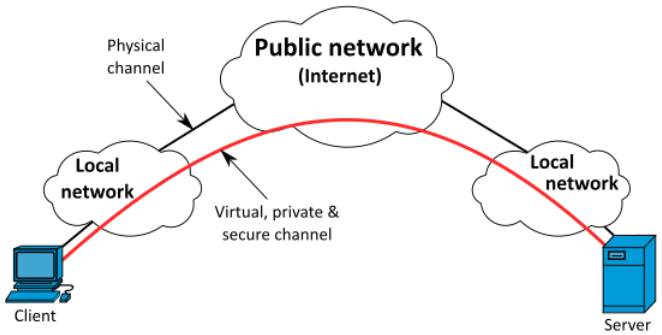
1.3. Introduction to VPN

A private virtual network (VPN) (“Virtual Private Network”, 2020) enables users to transmit and receive data over shared or public networks as if their computers were directly connecting with the private network. Thus the functions, security, and management of the private network can be enhanced through applications running on the computer such as a laptop, desktop, smartphone, through a VPN. Encryption is a usual part of a VPN connection, but not an inherent one.

VPN technology has been developed to provide access to corporate applications and resources byremote users and branches. To ensure safety, an encrypted layered tunneling protocol provides theconnection between private network sites, and the authentication methods of the VPN are used by

users of the VPN including passwords and certificates. Internet users may secure their connectionsto a VPN in other applications to circumvent geo-restrictions and censorship or to connect to proxyservers to protect their own identity and anonymous Internet location. However, some websites block access to known VPN technology to prevent their geo-restrictions from being circumventedand many VPN providers are developing strategies for tackling these obstacles.

A virtual point-to-point connection is established by using dedicated circuits or tunneling protocolsover existing networks. Some of the benefits of a broad-based network (WAN) can be achieved by a VPN available on the public internet (“Wide Area Network”, 2020). From a user's point of view, available resources can be accessed remotely in the private network.



A typical site-to-site VPN.

Figure 1.2: Virtual Private Network

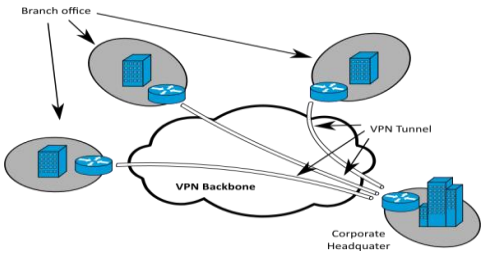


Figure 1.3: Site-to-Site VPN

1.4. Introduction of Botnet

The (“Botnet”, 2020) botnet is made up of several ISPs that are one or more bots each. Botnets can be used to carry out distributed DDoS attacks, steal data, send spam and enable the device andconnection to be accessed. Botnets can be used to send spam. The owner can use command and

control software (C&C) to control the botnet. The word "botnet" means "robot" and "network.""botnet". The term is generally used in negative or malicious terms.

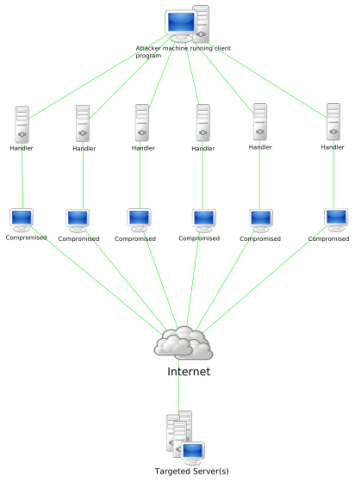


Figure 1.4: Botnet

1.4.1. Client Server Botnet

The architecture of the Botnet has evolved to avoid detection and interruption. Bot programs are traditionally built as clients communicating through existing servers. This means that the bot herder (manager of the botnet) can control all of his or her traffic from a remote location. Many recent botnets are now communicating using existing peer-to-peer networks. These P2P bot programs do the same things as the client-server model but don't have to communicate with a central server. Client-server model

Figure 1.5: Client-Server Botnet

A client-server model network that requires the services and resources of each client from centralized servers.

In the first internet botnets, a client server model was used to perform its functions. These botnetstypically operate via networks, areas, or websites of Internet Relay Chat. Infected clients have access to a default location and wait for incoming server commands. The bot herder sends commands to the server and transmits them to clients. Customers perform commands and report the results to the bot herder.

For IRC botnets, an infected client connects to the infected IRC server and uses the bot herder to connect to an existing channel designated for C&C. The bot-herder sends orders via the IRC serverto the channel. The commands are retrieved and executed by each client. With the results of theiractions, clients send messages back to the IRC channel.

1.4.2. Peer-to-peer

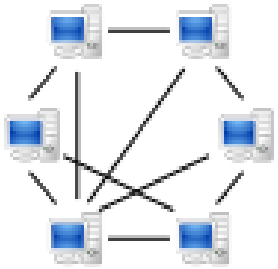


Figure 1.6: peer-to-peer (P2P) Botnet

A network of peer-to-peer (P2P) interconnected nodes ('peers') share resources without the use ofa central administrative system.

1.4.3. Zombie computer

In computer science, a zombie (“Zombie Computing”, 2020) computer is an Internet-related computer,which can be used for the malicious task of a certain kind under remote directions and has been affected by a hacker, computer virus, or Trojan horse. Zombie computer botnets are frequently used to spread spam e-mail and initiate denial-of-service attacks. Most zombie computer owners don't know how their system works. Because the owner is not aware of these computers, comparedwith zombies, they are metaphorical. An attack by DDoS coordinated by multiple botnet devices also looks like a zombie horde assault. Many computer users do not know that the bots are infectedwith their computers.

The process of stealing computer resources by connecting a system to a "botnet" sometimes is called "scrumping."

Command and control

Command and control (C&C) protocols from the traditional IRC approach to more advanced versions have been implemented in several ways.

- Telnet
- IRC
- P2P
- Domains
- Others

1.5. End-to-End Encryption

End-to-End Encryption ("E2EE(End to End Encryption)", 2020) refers to the encryption process at the end of the host. It is an asymmetric encryption implementation and thus guarantees a secure way to communicate data. This is the safest way for private and secure communication as data can onlybe read by the sender and recipient. Nobody else can encrypt the data that is passed through including the government or even the server.

1.5.1. Working of End-to-End Encryption

- The data communication between sender and receiver uses an asymmetric encryptiontechnique
- The transmitter pulls down the server's public key
- Then send encrypts messages to be sent with the recipient's public key
- Send the encrypted messages to the server
- The recipient will then receive encrypted server messages.
- The recipient then decrypts the messages with its private key
- Get the messages to read.

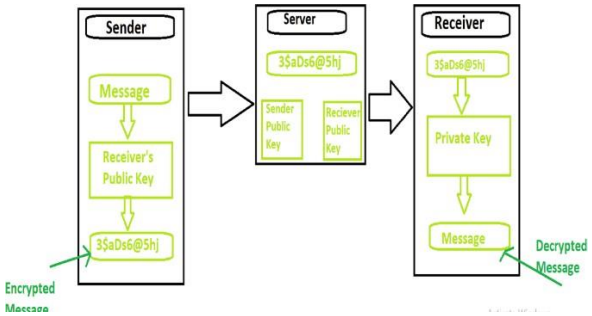


Figure 1.7: End-to-End Encryption

1.5.2. Applications

This method is extremely important when it comes to data protection, for example:

Negotiation and communication are very important and threatened. Military information to be safeguarded and to secure all communication Sensitive fields, such as health, minor information, etc.

1.5.3. Advantages

The communications approach is highly secure when traveling from sender to receiver. Thecommunications approach is highly secure when traveling from sender to receiver

- It allows the user to decide which data to encrypt.
- It also enables users to decide who can read their messages.
- The process of end-to-end encryption takes less time and resources and generally a smallfile size.

1.5.4. Disadvantages

For key storage, certain special devices are required.

- The data on endpoints is not protected

- Secure algorithm of encryption

1.6. Two-Factor Authentication (2FA)

2FA authentications (Linda Rosencrance, & others et. al., 2020) is a security process in which users provide two different authentication factors, which is sometimes known as two-stage verification or dual authentication. The 2FA authentication is a security process. This process is carried out sothat user credentials and user access resources are better protected. Authentication by two-factor provides a higher level of security than the Single-Factor Authentication (SFA) authentication method, where the user provides one factor – usually, a password or passcode. A user that provides

a password and a second factor, normally a security token or the biometric factor, like a fingerprintor facial scan, relies on two-factor authentication.

Two-Factor Authentication adds a safety layer for authentication by making it more difficult for attackers to access a person's devices or online accounts because it's not enough to pass an authentication check by knowing the victim's password alone. Dual-factor authentication is used to monitor access to sensitive data and systems, and 2FA is increasingly used by providers of online services to prevent hackers from using their user's credentials that stoles a password database or used phishing campaigns to obtain user passwords.

A Brute Force Attack: A Brute force attack is a hacking method that uses trial and error to crack passwords, login credentials, and encryption keys. It is a simple yet reliable tactic for gaining unauthorized access to individual accounts and organizations' systems and networks. The hacker tries multiple usernames and passwords, often using a computer to test a wide range of combinations, until they find the correct login information.

The name "brute force" comes from attackers using excessively forceful attempts to gain access to user accounts. Despite being an old cyberattack method, brute force attacks are tried and tested and remain a popular tactic with hackers.

1.7. Identity Management

Identity Management (IDM) ("Identity Management", 2020) is a framework for policies and technologies that ensures that the right user (in an enterprise) has appropriate access to technology resources. This management is also known as IdAM and access management (IAM or IdAM). InIDM systems, IT safety and data management are overarching. Identify, authenticate and manage access systems not only for individuals who will use IT resources but also for hardware and application staff. Identity and access management solutions have become more prevalent and critical during recent years, with increasingly stringent and complex regulatory compliancerequirements. It addresses an increasingly rigorous compliance requirement to guarantee adequateaccess to resources throughout increasingly heterogeneous technological environments.

Identity Management (ID Management) - or IDM - is the first organizational and technical processfor the registration and authorization of access rights at the configuration stage and then in operational phases for individuals, groups, and groups who have previously authorized access to applications, systems or networks. Identity management (IDM) is the control of computer user information. Such information includes the authentication of the user's identity and the descriptionof data and actions authorized to access and/or perform. It also includes managing the user description and how and through whom such data can be accessed and changed. Managed entitiesgenerally include hardware and network resources, even applications, in addition to users. The following diagram shows the relationship between IAM configuration and operational phases andthe difference between identity management and access administration.

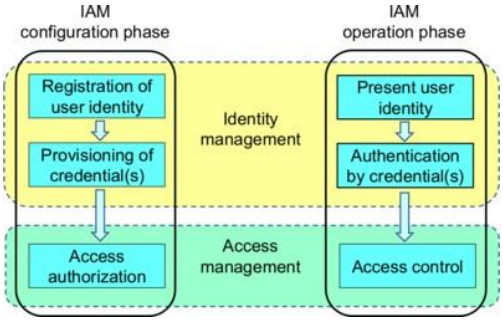


Figure 1.8: Identity Management

Access control is the enforcement of rights of access defined as authorization of access.

1.8. Opportunities and Challenges in VPN with IPv6

VPNs provide a secure solution (Sharma, V., 2020) to expose local services to the Internet thatare sensitive or vulnerable. For example, network management services, security systems, cameras, file servers, printers, etc.

Certain VPN features for IPv6 are not supported, including:

- IKEv2, while IKEv1 is not supported at the moment
- GroupVPN is not supported
- No VPN support for DHCP over VPNs.
- VPN Leaks

1.8.1. General Issues

IPv6 gives access to VPN problems, which require a certain amount of care. Since most people use routable addresses, NAT cannot hide the internal networks anymore. Firewall rules must therefore be carefully developed to allow remote network traffic. If the rules are too allowable traffic, LAN to LAN could reach without the VPN.

Ensure that, in such situations, the WAN-side firewall rules do not allow traffic on the remote LAN subnet and allow only traffic on the VPN interface. For instance, only remote LAN networkscan access the WAN of a VPN port on the firewall (s).

If the VPN is only used for connection, if the encryption level is not taken into account (for example, when services have already been encrypted and filtered appropriately), then IPv6 does not require a VPN and the traffic on the WAN interface can be allowed to go through the LAN.

1.8.2. OpenVPN

IPv6 on OpenVPN works on both routers and servers. The Windows client works and the latest versions of other notable OpenVPN clients such as Viscosity and Tunnel blick. Android and iOS clients are functioning as well.

Configure the OpenVPN server to use an IPv6 network for the tunnel network and appropriate routes as appropriate to get IPv6 on the OpenVPN tunnel.

1.8.3. IPsec

IPv6 is presently working with IPsec, but traffic in the tunnel cannot be mixed. In other words, IPv6 traffic can be transported only within a tunnel that contains IPv6 terminals, and IPv4 traffic can only be transported by IPv4 terminals. Both traffic types cannot be carried by a single tunnel.

1.8.4. Remote Access Mobile VPN Client Compatibility

pfSense® supports a variety of remote ("mobile") VPN settings that fit almost any potential customer. The table below shows which operating systems have compatible customers with some of pfSense's most popular remote access VPN settings.

Operating System	**Protocol**						
	Open VPN	IPsec					
		PSK	RSA	Xauth PSK	Xauth RSA	IKEv2 EAP MSCHAPv2/RADIUS	IKEv2 EAP TLS
Windows XP	3PA 12	3PA 3	3PA 3	3PA 3	3PA 3	?	?

Operating System	**Protocol**						
Windows Vista/7/8	3PA 12	3PA 3	3PA 3	3PA 3	3PA 3	Yes (7+)	Yes (7+)
Windows 10	3PA 12	?	?	?	?	Yes	Yes
Android <4	3PA	Varies	Varies	Bug	Yes	?	?
Android 4+	3PA 4	Varies	Varies	Bug	Yes	3PA 5	3PA 5
iOS < 9	3PA 6	?	?	Yes	Yes	?	?
iOS 9+	3PA 6	?	?	Yes	Yes	Yes	Yes
OS X < 10.11	3PA 2	?	?	Yes	Yes	?	?
OS X 10.11+	3PA 2	?	?	Yes	Yes	Yes	Yes
SNOM/Yealink	Yes	No	No	No	No	No	No

Table 1.1: Mobile/Remote Access VPN Client Availability

Yes = OS Native Client Available 3PA = Third-Party Client Required

Bug = Known problem configuration, follow the link for more detailsVaries = Varies by device model and vendor options

1.8.5. IPv6 Leaks

IPv6 leaks are one of the possible VPN leaks that may endanger a safety with a VPN.

In addition to a conventional IPv4 address, if the Internet provider (ISP) supports IPv6, the computer has an IPv6 address. And when using a VPN, if VPN does not have leak protection,the IPv6 address might accidentally leak.

Consequently, VPN suppliers have chosen for IPv6 leaks 1 in 2 possible fixes:

- VPN server/app custom code to avoid IPv6 leaks
- Completely blocking IPv6 (NordVPN is a well-known VPN that chose this option).
- The existing list of IPv6 support VPNs is fairly small, with most suppliers blocking theprotocol or routing it to the 'black hole' just like NordVPN.

The following are listed for VPNs with complete or partial IPv6 support:

- Perfect privacy
- Cyberghost
- AirVPN

1.9. Machine learning Algorithms

The study of computer algorithms is a process of machine learning, (“Machine Learning”, 2020) which automatically improves with experience and use of information. Artificial intelligence is seen as apart of it. Machine learning algorithms are designed to make predictions or decisions based on model-based sample data, called "training data," without any explicit programming. In many applications, like in medicine, e-mail filtering, and computer vision, machine learning algorithms are used where the development of traditional algorithms to perform the tasks required is hard or unfeasible.

A subset of machine learning is closely related to the statistics used in computer predictions; however, not all machine learning is statistical education. The study of mathematical optimization provides the fields of machine learning with techniques, theory, and application. Data mining is arelated field of study that focuses on the analysis of exploratory data via unattended learning. Machine learning is also referred to as predictive analytics in its application to business problems.

The traditional methods of machine learning are divided into three broad categories, depending onthe nature of the learning system 'signal' or ' feedback':

1.9.1. Type of ML Algorithms

Supervised learning: The computer is given example and output by a "teacher" to learn a general rule which maps inputs into outputs.

Unsupervised learning: The learning algorithm does not receive labels, which allows it to find its structure. Uncontrolled learning can be a goal itself (detect hidden data patterns) or a means ofachieving a goal (feature learning).

Reinforcement learning: A computer program interacts with a dynamic environment where a certain goal must be achieved (such as driving a vehicle or playing a game against an opponent). The program offers feedback, which is analogous to the fees it tries to maximize while navigatingits problem space.

1.10. Motivation for the Study

There are lots of misinterpretations about VPN. Today if required to create an anonymity then everyone recommends VPN. However, VPN is not a hacking product it is the most important resource for Enterprises to secure their infrastructure from Hackers or Botnets. Where VPN is used more for hacking purposes publicly rather than security. This issue creates the curiosity to deep dive into the VPN configurations and their technical pros and cons. The concept of VPN was developed to provide the local resources of an organization publically. However, the legacy VPNconfiguration does include security features and due to the complex network configuration, the network packet header

overlaps on its original values which like by the Hacking service providersand they market to product VPN for anonymity. Due to the Covid-19 pandemic, every organization is looking for VPN services so that employees can start work from home. That boost the requirement and due to the lack of security features organizations were scared to share internal resources for public access. As per the demand of society, this problem motivates me to work on it and give the best solution to the industry for the next generational VPN by proposing this securedVPN framework.

1.11. Need of Study

Because of the misinterpretation or lack of information, implementers have now experienced manyattacks and they never know that their VPN clients or servers will become Botnets members. They

also have the security of their data, infrastructure, networks, endpoints. The fully reliable solutionsare still lacking, which leads to hackers penetrating the VPN systems and turning them into zombies for malfunctioning. The problems grow every day and solutions providers continue to betested and tested. And in addition to very expensive, unguaranteed solutions.

1.11.1. Important Facts related to VPN

- **100% client confidentiality is offered by VPN providers**

Companies are simply claiming many false promises to market their products and they may vary according to jurisdiction at a national level.

- **A VPN may sometimes speed up the Internet**

However, it is a fact that our information always travels from the VPN because the Man in the middle of the attack is possible and that it might affect data safety. It is a myth that the Internet speed will grow.

- **VPN pay-free**

VPN is either payable or not because the paid VPN could give more IP from different countries and claim the anonymity of the client, in which case the people of Free VPN could create an anonymous IP, however, they can officially access the network activities and data breakage.

- **VPNs are firewalls that cannot be traced.**

One of the VPN's abuses is to make it easy for blocked service access via VPN if any IP or websitehas been blocked by a firewall. It is also known as a firewall bypass technique.

- **PPTP is the oldest VPN protocol.**

A Microsoft software engineer developed the oldest VPN protocol, PPTP Peer-to-peer tunneling in the nineties. This is the protocol opening up the VPN's way. PPTP is said to be the fastest protocol, but it was created for dial-up access. It has the lowest level of encryption.

- **No 100% anonymity guarantee for VPN**

Some VPN providers are anonymous to navigate the web, but they can't be completely anonymous.A user can be traced via the IP address if the VPNs do not request any personal information. Service suppliers claim that nothing is to be feared because the identity is never revealed.

- **Hack out a small option**

While VPNs helps to change the IP address, encrypt and hide data from third parties, however, cyber-attacks can still hack any system. Hacking is near to impossible with a strong VPN configuration so that efficient and trusted VPN services can always be used safely.

- **In some countries, VPNs are prohibited**

Thirty out of 196 nations are banning the use of VPNs. The authoritarian rules include China, Russia, North Korea, Vietnam, Syria, Saudi Arabia, Iran, and Burma. These governments prohibitVPNs to have full control over citizens' use of the Internet. For reasons such as imposing social values, maintaining political stability, and national security, any website that promotes opposing opinions will be reduced.

Few of the famous incidents have occurred because of the incompetence in cybersecurity policies;one of the main reasons is also the lack of knowledge. There are few incidences:

• Hacking of the VPN.

NordVPN is one of the most promising providers of VPN services around the world and confirmsthat in March 2019 their VPN services were hacked.

The OpenVPN provider Crypto Storm account on Twitter has a link in addition to the website certificate to an 8-channel post in the case of an individual hacker who claims full root access to NordVPN, Tor Guard, and Viking VPN servers. The attacker has been able to rob OpenVPN keysand settings files in the picture below the NordVPN hack. CryptoStorm.is said it could have permitted an assailant to decrypt traffic during hacking when stealing these keys.

• Hacker claims to have 18.000 Huawei routers using Botnet in July 2018,

• IoT promises to pay for damages

• Hacker claims that it is a pandora's box in October 2019

The current 2018 and 2017 threat landscape overview and comparison.

Top Threats 2017	Assessed Trends 2017	Top Threats 2018	Assessed Trends 2018	Change in ranking
1. Malware	⊃	1. Malware	⊃	→
2. Web Based Attacks	↑	2. Web Based Attacks	↑	→
3. Web Application Attacks	↑	3. Web Application Attacks	⊃	→
4. Phishing	↑	4. Phishing	↑	→
5. Spam	↑	5. Denial of Service	↑	↑
6. Denial of Service	↑	6. Spam	⊃	↓
7. Ransomware	↑	7. Botnets	↑	↑
8. Botnets	↑	8. Data Breaches	↑	↑
9. Insider threat	⊃	9. Insider Threat	↓	→
10. Physical manipulation/ damage/ theft/loss	⊃	10. Physical manipulation/ damage/ theft/loss	⊃	→
11. Data Breaches	↑	11. Information Leakage	↑	↑
12. Identity Theft	↑	12. Identity Theft	↑	→
13. Information Leakage	↑	13. Cryptojacking	↑	NEW
14. Exploit Kits	↓	14. Ransomware	↓	↓
15. Cyber Espionage	↑	15. Cyber Espionage	↓	→

Legend: Trends: ↓ Declining, ⊃ Stable, ↑ Increasing
Ranking: ↑ Going up, → Same, ↓ Going down

Table 1.2: A comparison chart of Top Threats in 2017 & 2018.

China and India are the world's biggest spambots serving counterfeit and malicious emails in vastnumbers. (ZAHARIA, A., 2020)

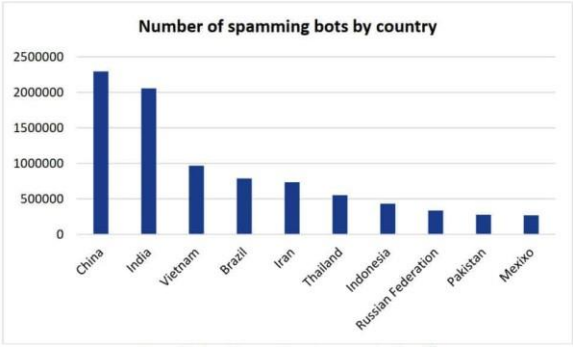


Figure 1.9: Top 10 countries Spamming Bots

2018-2019 Global Risk Exposure Information Security Survey

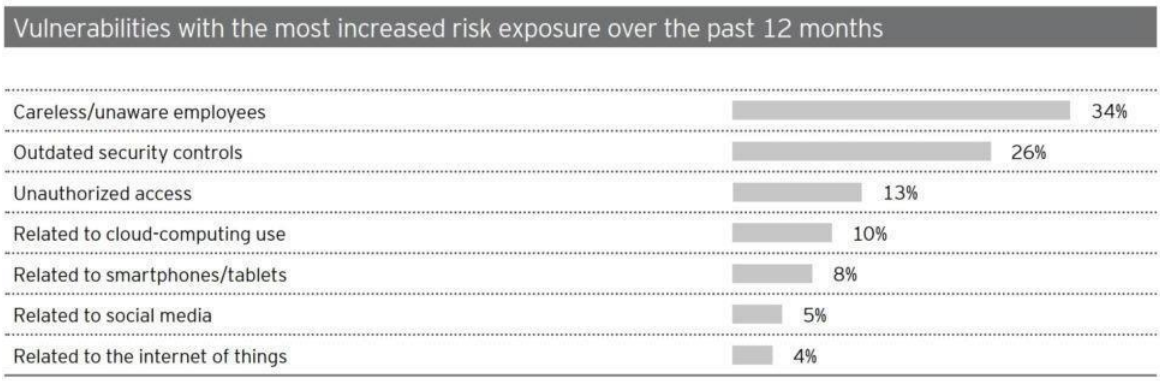


Figure 1.10: Risk Exposure Reasons

This study is sure to help society build on the latest technologies to create new dimensions of VPN securities that face identified or unexposed threats and challenges.

1.12. Scope and Contribution of the study

This framework covers almost all the organizations throughout the world to provide a secured environment where they can access their internal resources via the internet. The legacy VPN alsoprovides the access to internal resources but once the user got the access they can access all the internal network resources which might be confidential or required restrictions. So, this next generational VPN is overlapped with lots of security features and keeps tracking of all the activitiesof infrastructure and users which logged in the central server and the Machine learning algorithmssupport to analyze behaviors of the botnets, users, or complex infrastructure. To implement it hasbeen recommended to use open-source languages and platforms so that the expense is very less fordevelopers and the end-users. As it is based on open-source technologies, it is easy for troubleshooting, maintenance, and support. Overall, this contribution gives great relaxation to all the organizations in this IT world and in Pandemic Covid-19 there is a huge scope and demand from the society.

1.13. Research Methodology used in the Study

1.13.1. Benefits of Study

This research will help the whole world in particular the government, agencies, businesses, network solutions architects, developers of IDPs and policymakers, etc. These are of great help. Without a doubt, this study will explore the hidden challenges of cybersecurity and the latest technologies for implementing VPNs with IPv6. The

aim is to propose a security framework that respects confidentiality, completeness, availability, and authenticity. In addition, this study will provide a transparent, cost-effective, and reliable model to prevent the challenges and threats thatwill arise in the field of network security. This study would certainly protect the critical applications services, data, and digital identity against identified or unspecified malware from financial losses.

1.14. Key Concepts explain in study / Keywords used.

- Vulnerability assessment of IPv6
- Vulnerability assessment of VPN
- Identified and Unidentified threats from Botnet
- Common attacks on VPN using IPv6.
- Implementation of the various cryptography techniques to encrypt the data and client'sidentity.
- User Identity Management.
- Log Analysis and database management along with repository updates
- Machine Learning Algorithms for training and analysis of the dataset.

1.15. VPN Working and Common Protocols

A connection protocol is created in the device using VPN. This protocol establishes data communication from the device to the VPN server. There are a few commonly used VPN protocols, and each has benefits and drawbacks. Only a small number of the many communication protocols are supported by all VPN service providers. There are those who are safer than others. Some people move more quickly than others. Considering that you have a choice, consider your needs.

In Summary –

- **OpenVPN:** A relatively quick open-source protocol that offers strong encryption.
- **L2TP/IPsec:** This is also fairly common and offers speeds, but some websites that dislike VPN users easily ban it.
- **SSTP:** It provide strong encryption this protocol isn't very popular and doesn't have anything to promote it.
- **IKEv2:** Offers a highly quick connection that is particularly helpful for mobile devices, but has lower-quality encryption standards.
- **PPTP:** Extremely quick, yet over time has developed security weaknesses.
-

1.16. VPN Protocols Comparison

VPN Protocols	Encryption	Security	Speed
L2TP	256-bit	Highest encryption	Slow and highlyprocessor dependent
SSTP	256-bit	Highest encryption	Slow
IKEv2	256-bit	Highest encryption	Fast
PPTP	256-bit	Minimum Security	Fast

Table 1.3: VPN Protocols Comparison

1. **Layer 2 Tunnel Protocol (L2TP)**

L2TP (Layer 2 Tunnel Protocol) is the effective replacement for PPTP (Point to Point Tunneling Protocol) and L2FP (Layer 2 Forwarding Protocol) (L2F). However, it lacked encryption capabilities, it was frequently offered along with the IPsec security protocol. With no known weaknesses, this combination has been regarded the safest to date.

It's crucial to keep in mind that because this protocol use port number 500 (UDP), websites that reject VPN traffic can easily identify and block it.

2. Secure Socket Tunnelling Protocol (SSTP)

Since it has been rigorously tested and included in every version of Windows since Vista Service Pack 1, the Secure Socket Tunneling Protocol (SSTP) is a lesser-known protocol among common users yet is tremendously beneficial. Due to the use of 256-bit SSL keys and 2048-bit SSL/TLS certificates, it is also extremely secure. Additionally, because it belongs to Microsoft, no one else will be able to review it, which is both advantageous and disadvantageous.

3. Internet Key Exchange version 2 (IKEv2)

IKEv2, which was initially intended to be a tunnelling method, was created in association with Cisco and Microsoft. It uses IPsec to encrypt as a consequence. Its capacity to quickly re-establish broken connections accounts for much of its popularity among people who use it to deploy VPNs on mobile devices.

4. Point-to-Point Tunnelling Protocol (PPTP)

Point-to-Point Tunneling Protocol (PPTP), a VPN protocol, has been used for a long time. These VPN protocols are the oldest. This protocol has mostly lost favour because of serious security vulnerabilities, even if it is still used in select circumstances. It is useless because of a number of known flaws that have been used against it in the past by both good and bad people. Its only saving grace is speed. As I've said before, a connection's speed is more likely to deteriorate the more secure it is.

1.17. IPV6 Leak

One of the most serious vulnerabilities in VPN security is IPv6 leaks. Your system will have both a traditional IPv4 address and an IPv6 address if your Internet service provider (ISP) supports IPv6. The IPv6 address could unintentionally become public while using a VPN without leak prevention. As a result, VPN providers have decided between two possible IPv6 leak fixes:

The two potential IPv6 leak remedies have been chosen by VPN service providers:

1. Use custom code in the VPN server or app to prevent IPv6 leaks.

2. Complete IPv6 eradication (NordVPN is a well-known VPN that chose this option).

Why are some VPNs unable to handle IPv6?

For the following two reasons, many well-known VPN providers either don't provide or don't support IPv6:

1. There isn't really a need for it right now, so enabling it merely confuses matters more.

2. Prior to the widespread adoption of IPv6, there may be privacy concerns when utilizing IPv6 with a VPN on outdated OS systems (IPv6 address leakage).

IPv6 leak detection testing

Regardless of whether you're using an IPv6-compatible VPN or not, it's imperative to check for IP leaks. The quickest way to do this is to visit the ipleak.net website. When using a VPN, follow these steps to check for IPv6 leaks:

1. Obtain a virtual private network server account (VPN).
2. Visit the website IPleak.net.
3. Hold off until Unless and until every automated test has succeeded.

4. Check results for IP and DNS leaks.

1.18. Structure of the Thesis

The chapter scheme of the Research Synopsis is given below:

Chapter 1: Introduction

The first chapter of the thesis introduces the major topics used in the entire thesis. It includes a detailed introduction about IPv6, Botnet, Cryptographic algorithms like End-to-End encryption, (2FA) two-factor authentication, identity management, Machine learning algorithms, etc. It gives complete details about the study and opportunities and challenges using VPN.

Chapter 2: Literature review

In Chapter 2 it includes a literature review of the work done till now in relevant areas like VPN, VPN with IPv6, Mac Binding, DHCPv6, (SDN) Software-defined network, Customization of Network Packets, Cryptographic techniques like End-to-End encryption, (2FA) two-factor authentication, HTTPS, IPsec, Machine learning algorithms, log analysis, etc. At the end of this chapter, it includes the gap analysis which explains the real gap between present VPN configurations and the actual Secured Next Generation VPN configuration.

Chapter 3: Research Methodology

Chapter 3, includes the details about the research approach used in this thesis. It also includes the methods of data collection, research methodology used. Further, it extends to the initiated researchpart with quantitative analysis using survey questions and their analysis. It also explains the identification of the problem, hypothesis

testing, and the proposed framework of VPN including algorithms used to build the framework.

Chapter 4: Analysis and Interpretations

Chapter 4, includes detailed experiments for building, analysis, and interpretation of the framework. Here the explanation includes various experiments required for the proposed framework.

Chapter 5: Findings and Conclusion

Chapter 5, includes the relevant findings of the experiments used in chapter 4, further includes theconclusion, further investigations, reports, and recommendations.

CHAPTER 2: LITERATURE REVIEW

History of VPN Technology

Internet is a backbone of the VPN. The Internet is built from the fundamentals of ARPANET,a once American military computer network called as an ARPANET established by the US Defense Advanced Research Projects Agency in 1969. It builds up a fault tolerance computer network. The rest of the network will soon establish a new link if part of the network is destructed. The prototype of the Internet is usually considered. (ARPANET, 2020)

In 1983, ARPANET was divided into two parts, one ARPANET and the other MILNET, which was for military use only. A major contribution to Internet development was provided throughthe development of the local area network (LAN) and the Wide area network (WAN). The National Science Foundation is one of the most important organizations (NSF). ("Nation Science Foundation", 2020)

The NSF began establishing an NSFNET computer network in 1985. The NSF planned to create five supercomputers and an NSFNET network, which was a nationwide NSFNET for supporting and connecting scientific research and education with other networks. In 1989 the Internet name was used when MILNET (separated by ARPANET) connected to NSFNET. After all, ARPANET has disintegrated into the Internet the computer network of otherdepartments.

NSFNET made a major contribution to the Internet by opening it to society rather than government and research alone. In 1990, Advanced Network & Science Inc. was formed jointlyby Merit, IBM, and MCI. ANS was aimed at building a nationwide T3 backbone network andallowing 45 Mbps of data to be transmitted. By the end of 1991, the entire backbone network of NSFNET was connected to the T3 backbone network of ANS. ("Nation Science Foundation", 2020)

Trade organizations started entering the Internet in the early 1990s. They found that the Internet has great communication, information searching, and customer service potential. The internet has thus entered a new phase of marketing and business organizations have become a strong force for the development of the internet. NSFNET stopped functioning and the internetwas fully marketed in 1995. (ARPANET, 2020)

Moreover, The protocols TCP/IP boost the model of Internet. In 1974, a range of protocols including the famous IP Protocol and the Protocol on the Control of Transmission was issued (TCP). Both protocols cooperate. IP supports basic communication and TCP ensures reliable and stable IP communication.

Openness, which means that all TCP/IP and internet technology standards are open, is the mostimportant characteristic of TCP. The aim is to make any device capable of communicating witheach other, irrespective of the fabric that produces them, and one is to create an open Internet system. There are major reasons why the Internet has grown as fast as demonstrated in FIGURE2.1.

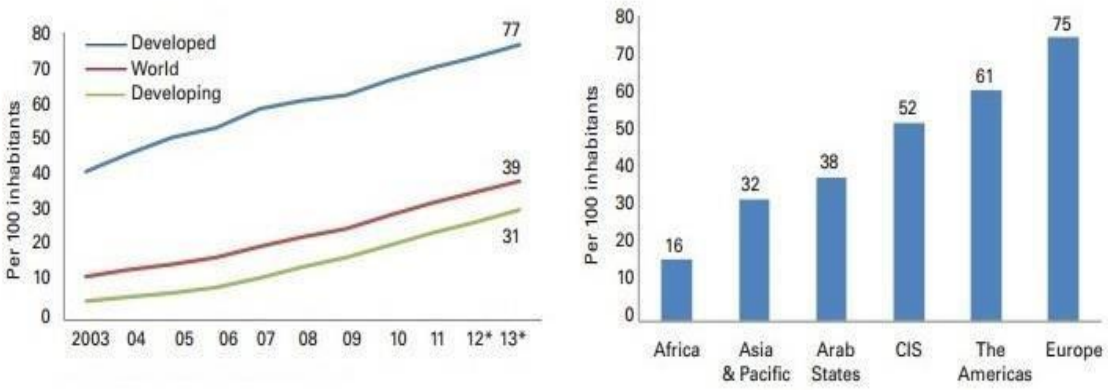


Figure 2.1: Internet users at development level, 2003-2013, and regional levels.

The Internet is getting bigger every day and as a result, not only are people sharing informationbut also people stealing information from others (nowadays, they called as hackers). Initially, the behavior of hackers was limited to macro viruses that slowed down or collapsed software and operating systems. Ultimately, the software sniffer and Trojan horse seemed to steal the password for users. Hackers today use high-tech and know everything about computers, software, hardware, networks, and operating systems. The attacks included death ping, hijacking sessions, and backdoor raids. In

particular, in the business sector, people focus moreon safety than ever before. (David, 1999)

The first way to protect against hackers is to create private networks, but that is too costly anddifficult to manage and maintain. Finally, VPN is another solution that can securely transmit data via the public Internet. VPN advances combine authentication and encryption using a tunneling protocol to build a nearly private network across the public Internet. That means thata large company network that is fully resistant to hackers can be developed and run by VPNs.The building of an expensive private network is also a good replacement. (David, 1999)

VPN is a remote access technology that creates an open-ended private network. For instance, when an employee goes to another city, he wants remote access to the corporate intranet serverresources. A VPN solution is for the intranet network setup of a VPN server. The VPN serveris connected to the intranet network by two network interface cards and the public network. Internet-connected personnel can find a VPN server and connect to the company intranet. Thecommunication between a VPN server and a client is encrypted to ensure data security. It maybe considering that data is in a special data connection for safe transmission through data encryption just as a specially established private network. VPN uses the Internet common linkso that only a virtual private Internet network can be called. In particular, VPN is essentially used to encapsulate the data communication tunnel that transmits data to the public internet through encryption technology. With VPN technology, whether at home or abroad, the user can be able to use VPN to access internal networking resources as long as there is an access tothe Internet. For this reason, VPN is so used in the company environment.

2.1 Early Layer 2 VPNs

Let’s introduce the examples of early layer 2 VPNs before describing existing VPN protocols.Here are the protocols X.25 and Frame Relay.

2.1.1 X.25

X.25 is the protocol proposed by the Telecommunications and Telegraph International Advisory Committee (It specifies that terminals and computers are connected to a packet network switched (PSN). The network switched to a packet is a network k in which the data packet picks the route to the destination. X.25 is a simple way to get the switched service of apacket. PSN used for connecting the remote terminal to the host system is traditionally used. These services provided every point to any point of connection for users who use the same service at the same time. The PSN can be entered through the X. 25 interface and distributed to various remote locations by different users' signals from the same network. In 1992, CCITTredefined the standards and improved the rate up to 2.048Mb/s, the X.25 interface can support

up to 64Kbps. X.25 is used today only on old networks in some developing world countries, and only for GPS tracking and wireless packet radio networks. X.25 is used today. (Sosinsky,2009)

The X.25 connects the Data Terminal Equipment (DTE) with Data Communication Equipment (DCE) by using Virtual Circuits (VC). VC s can be divided into two categories: Switch VirtualCircuit (SVC) and Permanent Virtual Circuit (PVC).

X.25, connects data terminal equipment (DTE) by virtual circuits with data communication equipment (DCE) (VC). VCs can be divided into two classes: Virtual Circuit Switch (SVC) and Virtual Circuit Permanent (PVC).

Use DTE to connect to the network is typically a computer, a terminal, and DCE can be a modem. For a user, X.25 could be viewed as an interface. The modem then transmits data viaa Packet Assembler Disassembler (PAD) gateway. PAD is a common mediating device between stations that send and receive. (Sosinky, 2009) FIGURE displays the common architecture

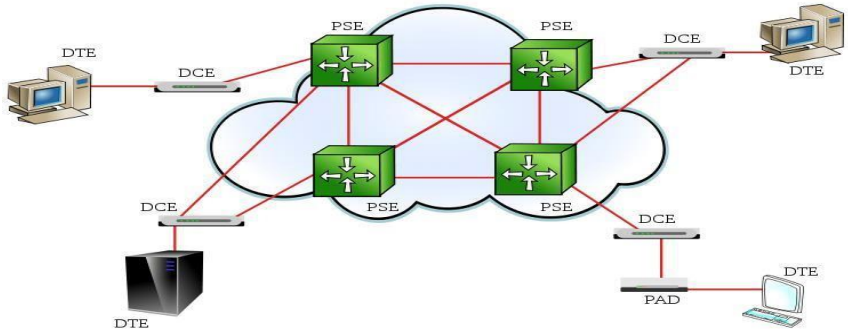


Figure 2.2: X.25 Network

This is how initially use a VPN. While it is a low-speed network, it provides a reliable connection for the proper and secure transmission of data.

2.2 Frame Relay (FR)

Frame Relay (FR) is an integrated digital network service development (ISDN). This is a packet-switching technology used primarily for Wide Area Networks (WAN) connections. Theold packet switching technology as mentioned before is regarded as a good replacement for X25. It reduces node processing time and improves network performance. (Paragraph 4, 2011)FR is the layer of the data link and protocol of the physical layer, all upper protocol is unrelated to FR. FR has no fix for an error fix, but detects an error and discards the error information only. It leaves the upper layers with an error correction function. Consequently, FR is cost- effective and less than X.25 overhead. It means the bandwidth of FR is higher and the delay less than X.25. (“Frame Relay”, (2020)).

Because FR is also a network for packet switches, it works like X.25. It divides the data into pieces called FR frames. The FR framework has a variable length and is dependent on the FRnetwork workload. FR is also working on the virtual circuit. It depends on whether it’s just a need of temporary or point-to-point connections. VC can also be SVC or PVC. While FR has no error-fixing function, FR has congestion management. The congestion controls include: Committed CIR used for regular-size throughput, Committed Burst Size (CB) for the highest permitted frequency, and

Excess Burst Size (BE) for additional frequency but not guaranteed (Sosinsky, 2009). Forward Explicit Notification for Congestion (FECN) shall be used to informthe DTE that the path from source to destination is congested. For transferring the frame opposite to FECN, the Backward Explicit Congestion Notice (BECN) is used. FIGURE showsbelow the basic structure of an FR network

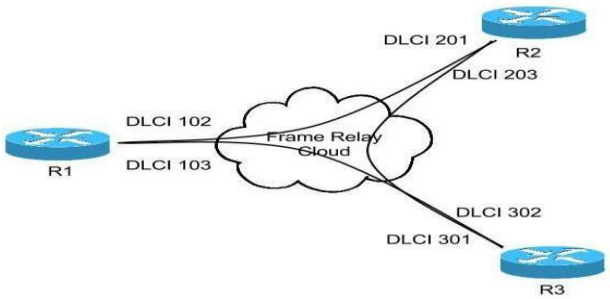


Figure 2.3: Logical Frame Relay

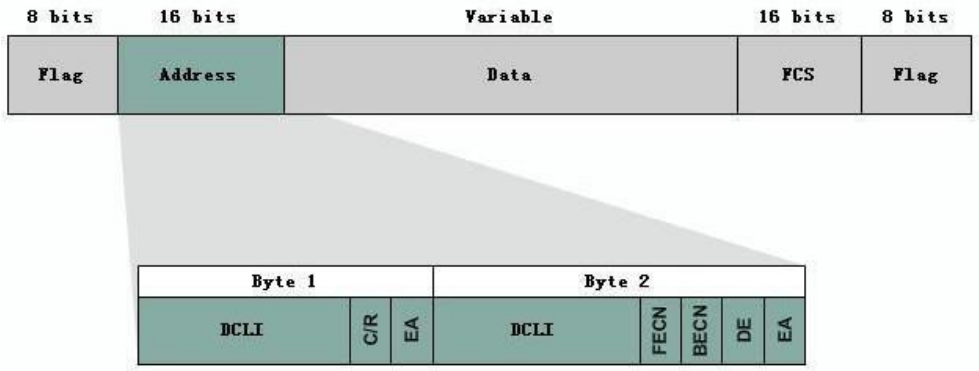


Figure 2.4: Standard Frame Relay Frame (Cisco, 2008)

The structure of a standard framework relay framework is shown in FIGURE 2.5 above. The largest frame size is 1600-bit, 16-bit address, 16-bit frame check sequence (FCS), 16-bit flag, and others data. Frame Check Sequence is for the error-fixing function, and before transmission, it will calculate the value. The terminal calculates a value and compares it to FCSafter the frame arrives at the target. FR transfers it to the upper layer if they are the same. Theframe is dropped if they are not the same. The next thing is the Identifier of the link data (DLCI). DLCI shall be stored in each data frame's address field. It is used to say how the dataframe is transmitted. DLCI has local importance only, which means that in the FR network these values are not always the same. The Virtual Circuit (VC) is only identified by DLCI. DLCI has no sense in

signaling a connection. Two different VC-connected devices can display the same connection by using different DLCI. (Cisco, 2008)

FR is a good replacement for X.25 because it uses PVCs and provides a point-to-point connection with lower and higher bandwidth. It is a good choice for speed improvement and good safety.

2.3 Current Layer 2 VPN

The layer 2 protocol used for the VPN are as follows:

2.3.1 Multi-Protocol Label Switching

A network change technology was proposed in 1997 by the Internet Engineering Working Force (IETF) as the Multi-Protocol Label Switching (MPLS) technology. It mainly adds connectivity in traditional IP networks to improve the network's exchange rate by using the openness and flexibility of IP technology. The complexity of the network will also be reduced.This is why MPLS is becoming increasingly popular. (Chen, 2009)

Label Switching Router is the basic unit of the MPLS network (LSR). Label Edge Router (LER) is the edge of the network and LSR is the core node of the network. The IP packet entryand exit of the LER node is the responsibility of the MPLS network; the high-speed packet exitof the LSR node is responsible. The defined class for forwarding equivalence is each packet that enters the Mpls network (FEC). Each FEC encodes a value called a label that is short but of fixed length. The Label for the Distribution Protocol between devices (LDP). The Routing Protocol between LSR and LER, LSR and LSR remains in operation for the MPLS. Then devices decide the label switching path based on the routing information (LSP). (Heydarian, 2012)

When an IP packet enters a large LER network, the incoming LER scans the packet header andsearches the routing table to find the destination of the LSP. Finally, the appropriate LSP labelis inserted into the header. The packet is forwarded to the LSP which is shown by the label after this procedure. The network node transfers the packet

based on the label of the IP header,so it won't have to repeatedly check the routing table. The LSP strips the label of the packet ifthe packet enters LSR. Following normal IP routing, the packet is then transmitted to the destination. (Chen, 2009). FIGURE 2.5 displays the MPLS network structure.

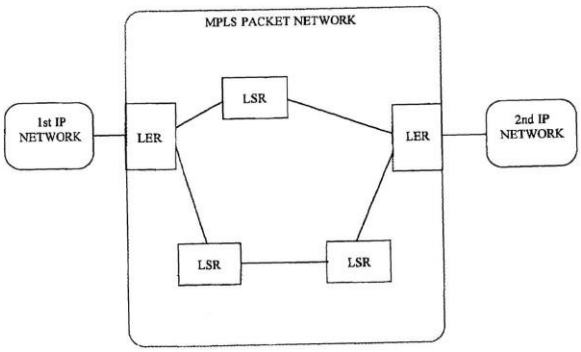


Figure 2.5: MPLS structure

MPLS is highly suitable for VPN, as described above. The performance of VPN is improved and the transfer of data cannot be seen in the LSR network. They are all labeled in FEC, so MPLS is popular in existing networks.

2.4 LAYER 3 VPN

2.4.1 Generic Routing Encapsulation

GRE is a protocol that encapsulates one protocol over another. It was released by IETF in 1994.(Hanks, 1994) Not only the IP protocol but other network layer protocols are supported by GRE. It allows a payload packet to be used in any kind of protocol, and it can be embedded inany data packet of other kinds. (Worster, 2005)

There are three parts to the GRE protocol datagram format: Delivery Header, GRE Header andPayload Packet. (Hanks, 1994) FIGURE 2.6 shows the GRE protocol stack.

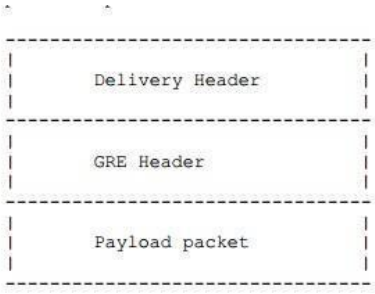


Figure 2.6: GRE protocol stack (RFC 1701)

The payload packet, viewed from FIGURE 2.6, is the data the user is to send and the data is tobe encapsulated. To set up, maintain, and end the tunnel, a GRE header is used. The load packetis enclosed, the GRE header is added, then both parts are placed in the IP data field and transmitted finally via IP. The delivery protocol that is used for transmitting the GRE header embedded and the payload is added to the Delivery Header. For instance, IP is the most frequenttransportation protocol, usually used is GRE's IP protocol. (Zheng, 2016)

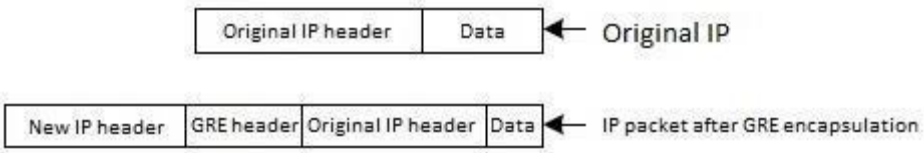


Figure 2.7: Process of GRE encapsulation

Figure 2.7 illustrates the process of GRE encapsulation. If an IP packet needs a tunnel, add a GRE header in the first place, then add a new IP header based on the tunnel's IP address, andthen forward it through the new IP header. The process of decapsulation of the GRE tunnel is the opposite process. (Zheng, 2016)

GRE is a fundamental VPN tunneling technology. It's an original VPN model, simply change the algorithm, add authentication and other security features.

2.5 An Analysis of Literature Survey

Many papers were written by famous researchers on the topic analyzed. Their approach was related to the use of VPN in various domains, their types, the challenges and opportunities forimplementing an IPv6, and behavior analysis of botnets and other

malware, etc. Thus, the research gap and the lack of solutions implemented so far can be identified. In the literature survey, the major keywords or technologies are listed below:

2.5.1 National Informatics Centre (“NIC VIP Policy”, 2019). GOI proposed in this paper – NIC, M.I.T, the Guidelines to Connect NIC Hosted Internal Server to Reduce Unauthorized User Exposure Potential. This policy defines the policy of remote access to NICNET servers from the web via IPsec or SSL.

2.5.2 Cisco.com (Cisco, 2019). The procedure to connect offices through VPN tunnels is explained in this document from Cisco, one of the world-class companies. They emphasize IPsec and other safe methods for connecting VPN customers with office infrastructure.

2.5.3 Meta Networks (“SDP-vs-VPN”, 2018). The research paper was published by one of theother renowned networks and product providers to analyze the best practice in defined softwarevs. virtual private networks. In this paper, they discuss SDP's compelling alternatives to traditional VPN that enable organizations to standardize the security of remote access for all users, to scale them more economically, and to reduce the risk of attacks.

2.5.4 Mohammad Taha Khan (Khan, 2018). In this paper – The author suggests that a substantial part of the VPN ecosystem may contain providers seeking the benefit of their privileged position between the Internet and its customers. Prior work shows, in particular, theinterception and manipulation of traffic by large numbers of viewpoints. Many of the VPN services believe based on standardized protocols and customer software, which decrease providers' chance of carrying out malice (e.g., TLS interception or sudden client traffic routingthrough other customers). Finally, it was identified that all types of VPN providers often have poor defaults, resulting in unintentional data leakage particularly in the event of a tunnel connection failure. Even in 2018 safely using a VPN remains a task that is mainly beyond the aficionado's reach, and no VPN is the perfect security solution at least.

2.5.5 Guard (guard, 2019). In this paper – The implementation of a secure VPN on an Androiddevice was identified as a complex task that requires deep changes to the device's software platform. Inside Secure has developed a VPN customer that meets

the most stringent security demands from carriers, companies, and government, and in cooperation with leading Androidoperators and dozens of year investments.

2.5.6. Byeong-Ho Kang: (Kang, 2009). In this paper - The attack described threatens all VPNsthat use preshared authentication keys and accept VPN connectors from anywhere such as travel users' access. The authors also proposed policies for guidelines to give the company company's corporate network virtual private network IPsec remote access connections.

2.5.7. Muhammad Ikram: (Ikram, 2016). This paper – The discussion discussed an Android VPN Permissions-enabled apps Privacy and Security Risk Analysis. Through the VPN permission to offer censor circumvention, to support the company's customers, and to enhance Online Security and Privacy, Android app developers benefit from native support. However, even with millions of mobile users worldwide installing Android VPN-enabled apps their operational transparency and potential impact on user privacy and security, even for the technologically knowledgeable, is "terra incognita." The authors presented several static and dynamic methods that permit us to perform a thorough analysis on Google Play of VPN- enabled apps. The study consisted of tracking services and malware on VPN app binaries to artifacts implemented on a network basis by these applications. Our extensive testing has allowed us to identify instances of VPN applications that incorporate third-party tracking services and implement abusive practices, for example injecting JavaScript, redirecting ads, and even TLS interception.

2.5.8 JAYANTHI GOKULAKRISHNAN: (Jayanthi, 2014). In this paper - the different risks and vulnerabilities in the VPN are explained and the various safety protocols in VPN are explained. Three security protocols and their features were the main focus of this paper.

2.5.9 Sanaz Rahimi: (Sanaz, 2017) This paper – I have been talking in INDUSTRIAL CONTROL ENVIRONMENTS about SECURITY ANALYSIS OF VPN CONFIGURATION. A stochastic VPN model provides a powerful framework to analyze the effects in industrial control environments of different configurations and operating modes on VPN security. Model simulations contribute to the quantification, evaluation, and security compromise of security protocols, providing a basis for the safe operation of VPNs.

2.5.10 Thanh Bui: (Thanh, 2019). The authors discussed customer vulnerabilities in Commercial VPNs in this paper. They examined the safety in terms of setting up or instructingtheir users to set up, the popular commercial VPN providers for VPN clients. They have beenworking commonly on Windows, macOS, and Ubuntu VPN protocols and software. The clientconfigurations of most protocols and clients have identified vulnerabilities. The vulnerability allows network attackers to impersonate MitM or the server and thereby obtain the original network traffic of the victim. The vulnerabilities Local attackers can also take advantage of vulnerabilities to steal VPN services user credentials. Give guidance on how these vulnerabilities can be fixed. The major concern is that security deficiencies are not always hidden deeply in code or encryption, either accidentally or deliberately. Instead, simple

configuration errors, bad instructions, unsafe default values, and the absence of broken legacycharacteristics can lead to widespread industry failure.

2.5.11 J. BALU: (Balu, 2014). In this paper, the authors discussed the safe transmission of datathrough WiMAX networks in a real-time environment using VPN technology. For both MAClayer security and IPsec, the secure data transmittal over the WiMAX network using VPN technology using testbed experiments is evaluated. While IPsec provides strong data security via IPsec tunnels, the QoS performance can be improved by using VPN MPLS technologies both for wired and wireless networks. No articles have however reported actual experiments oractual overhead measurements on IPsec. This paper presents results for further analysis and comparison from testbed experiments that are both practical and theoretical. Modified IPsec can be combined to support mobile WiMAX networks based on existing research.

2.5.12 E Ramadhani: (Ramadhani, 2018) The author talked in this paper about the communication on anonymity A comparative study. VPN and Tor. VPN and Tor differ based on their secure communication. VPN and Tor have a very similar purpose in maintaining the online anonymity of internet users and evading firewalls. It completely depends on a user the way wants to use VPN and Tor services. Each one has its advantages and inconveniences. It can be combined to make communication more secure. One thing that must be noted is that anonymity is not completely guaranteed. Use it wisely based on necessity.

2.5.13 J. Myles Powell: (Powell, 2010) In this Paper – The author talked about the impact ona company's network of Virtual Private Network (VPN). The use of a VPN

means that the network load and time delay are significantly increased. This is however a small cost to pay for a virtual private network's security and privacy. VPN is the most efficient and versatile form of safe long-distance communication. Additional network load requires more bandwidth. A VPN may require upgrading computer hardware or additional hardware. In the absence of development and extension of network resources to meet new VPN requirements, companies may have slow response times in e-mail, file delivery, and database inquiries. Model research revealed a 446 percent delay in the query by using a VPN for conducted database transactions.In e-mail and FTP transactions, a significant delay is also added. The early costly solution forprivate networks was leased lines and frame relay networks. Its higher costs and higher hardware requirements lead to VPN technology spreading. PPTP and L2FP protocols have been developed and the VPN technology has been integrated. The requirement for greater security led to IPsec technology being integrated into the existing VPN framework. This also shifted the focus from layer 2 to layer 3 of VPN technology. Today users have a safe, cheap, and convenient virtual private network to access resources remotely.

2.5.14 K. Karuna Jyothi: (Karuna, 2018). This paper – the study is based on Virtual Private Network (VPN). The user can see the secure connection as private network communication, although it is via a public Internet. They classified all the various types of VPNs and noted theflexibility of which systems the customer can use. VPNs can offer various algorithms of encryption, authentication, and integrity. The company can develop a security profile for its offices and select the most suitable VPN solution. They examined the various protocols used in VPNs carefully and observed that no standard was adopted by a clear majority as a result ofVPN technology being new. VPNs are still young and the full potential of VPNs is still to be used. For the future, VPNs are promising for secure Internet communication. The VPN industryis expected to be a very large market in the years to come. It is important that the selected standards match the needs of the customer and that they remain flexible. VPNs are a highly secure, flexible, low-cost communication tool. In the coming years, the development of this new technology could well define the standard for secure Internet communication.

2.5.15 Manal M. Alhassoun: (Alhassoun, 2016). A survey on IPv6 deploying was carried outby the author in this paper. The protocol was found to replace IPv4 in its entirety, but it will take a few years for us to migrate fully. Therefore, a lot of research has been conducted to findstrategies and mechanisms to transmit IPv6 incrementally so that it can maintain interconnections between the protocols and enable them to coexist without problems. The change is quite unavoidable. Implementing and deploying IPv6 is indeed a difficult, risky andexpensive task; but its implementation tools and methodologies can be much easier and more effective with good planning and the optimum choice.

2.5.16 Dr. Sandeep Tayal: (Tayal, 2017). The authors discuss issues of implementation in IPv6network technology in this paper. The few reasons why IPv6 is late in taking the device is thatit is not compatible, i.e. layer 2 devices can work with no or some modification. However, because of their costs and various technical issues, industries and companies do not want to upgrade their devices. Another reason is the use of IPv4 addresses by the backbone routers. It'salso required to change their routing tables.

2.5.17 N. Sekar: (Sekar, 2017). In this paper – the authors conducted a study in networking onthe ipv6 Internet Protocol. The idea of the IPV6 network is clearer than the IPV4 network. The contribution of this related project is to find a way to reduce the time for transfer, including theloss and jitter of packet delays.

2.5.18 Amit Tambe: (Tambe, 2019). In this paper – the author(s) work with Scalable VPN- furthered Honeypots on detecting threats to IoT devices. They proposed the design, using VPNconnections, of a general honeypot framework for IoT devices. They collected true and high- interaction attack traffic through low-effort exposures on multiple IP addresses of COTS devices that offered geographical diversity. The proposed 12 COTS devices frameworkincluding IP cameras, printers, smart plugs, and other intelligent devices. The proposedframework These devices have been exposed to 44 IP addresses in nine countries in 15 cities. Then several precautions were taken to detect and mitigate the spread of malware. They did not directly detect compromise by the COTS devices, but instead, rely on traffic analysis. They also suggested and implemented two methods to analyze live traffic and used traffic methods captured by the honeypot. There was 5136 successful and malicious attempts at logging outbound connections based on the analysis of the traffic captured. These links have been attempted in 109 countries to achieve 4642 different IP addresses. It

has been attacked throughcommands to 97.5% of outbound IPs via live traffic analysis methods.

2.5.19 Srinidhi K S: (Srinidhi, 2014). In this paper, the authors propose 'AUTOMATIC BANDWIDTH IPV6 TRANSITION.' In IPV4 to IPV6 and also with IPV4 and IPV6, tunnelingis the best option for migration. Only near customers should install dual-stack routers and the already existing IPV4 device network remains untouched. It helps to transfer traffic in IPV4 through encapsulation and decapsulation successfully in the IPV6 network. In the case of ciscorouters, the disadvantage of the existing manual bandwidth assignment is that customers under-utilize or over-use the bandwidth. A new automatic bandwidth management algorithm is proposed to overcome this drawback, where the priorities of the related customers have not been taken into account and every service shall provide a minimum and maximum bandwidth,which will ensure a minimum bandwidth for each customer.

2.5.20 Ting-Ting Zhang (Zhang, 2012). In this article, possible ways of measuring and analyzing IPv6 network performance are explored and advanced by a thorough analysis of analysis parameters and methods for network performance. In this research, experiments are conducted to measure the roughness of IPv6 network performance, in which Fluk is employed for the analyzer of protocols and D-ITG as a grouped communications source. Experiment results are recorded. In addition, an IPv6 dual-stack and IPv6/Ipv4 tunnel testing device calledNPT4/6 is designed and then applied. Major network-internal network performance indexes

are examined, like RTT, loss rate, route capacity, etc. The results of the measurement demonstrate the current IPv6 tunnel performance suspect.

2.5.21 Vasile C. Perta (Perta, 2015). This paper - a glimpse through the VPN Looking Glass: IPv6 and DNS Hijacking in Commercial VPN customers. It was found that almost all VPN services are covered by a serious vulnerability, IPv6 traffic leakage. In many cases, the wholeof the IPv6 traffic of a customer was measured via the native interface. Two other DNS attacksthat allow us to access all victims' traffic revealed another safety screening. The most alarmingsituation is that people use VPN services to protect themselves against surveillance in repressive regimes. Users that believe that their data and online activity will be fully disclosed, in these cases, that they are anonymous and secure. These monitoring systems can take the roleof any of our opposing models in countries that have government network infrastructure. VPN

services are exposed to the major misinformation end users so that they find it difficult to properly disagree with real facts with vague and audacious claims typical of product campaigns. For example, in top positions, a simple Google query such as Tor vs VPN returns several web pages that are not affiliated in any way to the Tor community. Even one from a business VPN service provider site says that a VPN service is probably a better option for Tor,to achieve better privacy protection.

2.5.22 Tina Sharma (Sharma, 2013). In this paper– the authors introduced IPsec to IPv6 networks as their statistical results. The diagram shows the minimum throughput generated bythe IPsec tunnel created in the network in the graph. The graph shows the minimum traffic thatcan be transmitted from source to destination. It was found that the minimum rate of performance for User A was achieved at 312 bytes/sec, after analyzing ping results for all users.At 5720 bytes/sec by User F, maximum output is achieved. It specifies the maximum traffic from source to destination.

2.5.23 Geethamani G.S (Geethamani, 2018). The paper – See Secure VPN for Moving Target Defense based on Network IPV6. Three key network movement defense features were described. A moving defense strategy based on a new mobile IPv6 is designed to constantly change the IP addresses, making it difficult to find attackers. 1. Null network delay extra 2. Nullpacket loss 3. Cutting-edge 4. Versatile Overhead signals: Every IP change round will require 2 message transmissions per MN (BU and BA) of 158 bytes each (using IPsec). A new movingIPv6 network defense strategy aims to continually change IP addresses to make it difficult forattackers to find them. There is 24 bytes of overhead for each data packet because IPsec is used(ES).

2.5.24 Prabhu Thiruvasagam (Thiruvasagam, 2019). The paper – IPsec: Performance analysisin IPv4 and IPv6 was discussed in this article. IPsec's performance throughput using various combined encryption algorithms, encryption algorithms, and authentication algorithms in the way the IPsec implementing standards in NDS/IP networks recently suggested. The Strongswan Open-Source-Tool for the implementation of IPsec has been used and the appropriate configurations and setup details have been presented. In three different cases, the performance of IPsec was analyzed based on the chosen algorithms. The results showed that AES-CTR performs better than AES-CBC and 3DES with AEAD algorithms AES128GCM16and SHA1 performs better than SHA256. it's planned to examine the performance of algorithms from AESXCBC extensively and, in particular, why they are abnormal.

2.5.25 Hyung-Jin Lim (Lim, 2006). In this paper, depending on the situation in the network, the IPv6 deployments use various trans mechanisms. A mobile network should therefore be able to connect to a home network securely. In safe connections to a corresponding home, thenetwork is an important factor in the correlation between the mechanism of transition used by existing networks and transition mechanisms supported by the mobile network. During the transition to IPv6, the VPN scenarios for mobile networks were analyzed. To establish connectivity according to the VPN models at every transition phase, performance costs were also evaluated. Moreover, through numerical analysis of the NEMO VPN model and IPv6 transition environment, this paper analyses factor affecting performance. As presented, after careful evaluation of not only security vulnerabilities but also performance requirements, a NEMO VPN that creates a hierarchical or sequential tunnel should be proposed.

2.5.26 Rahmadani Hadianto (Hadianto, 2018). The architecture of the SDN, which divides control plane functions and data plane functions, provides benefits on different sides; easy network systems operation by programming network functions, cost efficiency in developing and improving the network; and opportunities for adapting new technological developments tothe SDN. However, research on the security system of SDN is currently not very developed, as demonstrated by the lack of research that addresses the defense aspects of the change in attacks on the security of the SDN Network. Based on the review paper, some aspects of the SDN safety system, in particular in Botnet attacks, must be addressed. With this problem, its required to develop different security methods to handle an attack variation. The handling of botnets with honeypot systems is one of the most potential safety methods in the processing ofbotnets. Where honeypot proved capable of handling and learning botnet on the traditional network.

2.5.27 Chigozie-Okwum C .C (Chigozie-Okwum, 2019). In this Paper - Machine learning techniques are used for botnet identification. The use of machine learning algorithms (MLAs)to identify patterns of malicious traffic is a trend of the latest in network-based botnet detection.Machine learning methods mainly rely on botnets to build distinctive patterns throughout network traffic and to effectively detect these patterns using MLAs. This detection class promises to generalize the knowledge of

malicious network traffic from available observations and thus prevent pitfalls of signature detection approaches that can only detect known traffic anomalies.

2.5.28 Andrew Thompson (Thompson, 2017). In this paper – Three research papers to improvebotnet detection have been analyzed and evaluated. The high detection rates found in every research paper demonstrate the scientific merit of each approach to botnet detection. The paperrecommends that future research should be focused to remove the problem of packet loss documented in the conclusions of the researchers. The combination of Soniya and Wilscy's filter module and the Singh et al. documented 'Traffic sniffer' module may serve as an effectiveway to do this because Soniya and Wilscy had no packet loss. If this combination succeeded ineliminating packet loss, the next logical step would be real-world testing. Research on the ability to detect a wider array of Peer-to-Peer botnets should be conducted. The results providedby Yahyazadeh and Abadi are not discredited, as they focused on the detection of IRC, HTTP,and P2P botnets in the same detection method.

2.5.29 Vorsu Yadigiri (Yadigiri, 2017). In this Paper - A survey based on anomaly and community detection on botnet detection. They propose a new two-stage method for botnet detection. The first stage covers the network traffic through a sliding window and monitors network abnormalities. They proposed two methods of anomaly detection, each based on largevariations in flow and packet-level data. An anomaly may be represented as a set of interactionrecords for both anomaly detection methods. In the proposed method for the detection of compromise nodes once cases of abnormalities have been found. This is based on ideas from social networking community detection. However, it has been decided to develop a refined measure of modularity suitable to detect botnet. In addition, refined modularity addresses certain modularity constraints by adding regularization terms and combining data for key interaction measures and SCGs.

2.5.30 Shahid Anwar (Anwar, 2018). In this paper - A serious threat to Android devices. Discussion about Android Botnets. Botnets from Android are harmful to Android devices. Mobiles have become a soft target for possible attacks thanks to their popularity. This survey

sought to detect the true threat of Android botnets. They conducted an extensive survey and detection techniques of existing Android botnets. The detection techniques, such as static, dynamic, and hybrid detection techniques, were categorized according to

their environment. There are organized limitations and benefits in existing Android botnet detection techniques. This is the latest organized survey on Android botnets and their detection techniques to the bestof our knowledge.

2.5.31 Anup Girdhar. (Girdhar, 2010). In this paper – The Media Access Control (MAC) (a unique number of 48 bits in hexadecimal values) was mentioned. It is installed by themanufacturer in the chipset of any network card. The MAC address instead of the IP address for the exact machine used to transfer any data in the network. However, it also said that a change in the configuration of the operating system may also spoof the MAC Address. Also mentioned in the paper are few solutions to mitigate MAC spoofing.

2.5.32 Kartik Giri and others (Kartik, 2020). This paper Briefs the various End-to-End Services encryption techniques. The strengths and the vulnerabilities of each encryption technology arepresent. Where there may be a lack of availability for one technique, another may be poor distribution. The only way to determine which one is superior is to evaluate and compare different methods among all the techniques used in the modern world. Therefore, they have todecide as to what kind of data they want to secure for use in the encryption techniques. To addup all the above strategies, continuing encryptions are helpful. Regular new methods of encryption advance, so customary, fast, secure encryption processes work with higher security rates consistently.

2.5.33 De Cristofaro and others (Cristofaro, 2013). In this paper, A comparative two-factor authentication (2F) technology exploratory study was presented by the author. In a pre-study interview with 9 participants, authors first reported on the aim of identifying popular 2F technologies and also how, when, where, and why they are used. Now designed and administered an online survey to 219 Turkish mechanics to measure the usability of a few popular 2F technologies: unique token codes, once-on-time PINs received via SMS or e-mail, and smartphone apps. They also recorded contexts and motivations and investigated their effecton the sensitivity of the different 2F technologies. The participants considered who employedspecial 2F technologies

because they had to or wanted to. To estimate the usability, confidenceand cognitive effort are the three key points defining 2F realizability, in a study of the exploratory factor to evaluate a set of parameters, including some suggested by previous work.Finally, it has been identified that differences in 2F use depend, rather than the current technologies or contexts, on the individual characteristics of individual individuals. The few features, for example, age, sex, and informatics, and gained some insights into user preference.

2.5.34 S. Alspaugh. In this paper, detailed data describing the log analysis process were provided by the authors. Although several user system administrator studies were conducted, there were few if any quantitative reports on user behavior trace on the detailed level of an actual log analysis system. Furthermore, it provides high-level quality survey data. Two important sources of information are used to inform product design, guide user testing, build statistical user models and even create intelligent interfaces that advise users to increase analytical capacity. First, the main observations were summarized and followed with an appeal for current toolmakers and future researchers.

2.5.35 Common Vulnerabilities Exposed in VPN – A Survey, In this paper, study of COVID-19 Pandemic. In Pandemic , Internet traffic has been increased by up to 90%. Workfrom-home culture is initiated by almost every organization. The technology adapted to access the Enterprises Intranet is VPN (Virtual Private Network). Infrastructure administrators implemented/ updated VPN with the latest versions along with the security scripts to access Intranet. However, the contingencies faced by the organizations are out of their scope. Now VPN security is a big challenge for almost every organization. The Veracity is that no one claims the full prove security system in their Infrastructures. The latest Vulnerabilities have been exposed and indexed in context to VPN Hardware's/ Software's/ Configurations and Implementations. In this paper, it has been decided to analyze the exposed VPN vulnerabilities, along with the ongoing issues which have not been listed to date through the survey. The mitigation policies have been proposed based on observations.

2.6 Gap Analysis

The main gaps in the research carried out until then when studied various research papers, URLs, and other materials relating to the topic. Some papers only talk about the VPN, IPv6, and the Botnets functionality. Few researchers, however, talk about

malware's commonbehavior and other IPv6 and VPN attacks. The problems associated with the combined VPN problems implemented in IPv6 are not expressed. Various attacks related to Android VPN andIOT IPv6 were also found. The great disadvantages and research gaps show that many devicesstill fail to comply with IPv6 even after decades. Many mobile applications are incorporated into the VPN services and Indian devices are misused as zombies. Due to the unknown solutions for detecting botnets working on IPv6, there are possible risks for the entire infrastructure. In particular, companies require data confidentiality and authenticity with theircritical application servers/sites. The latest vulnerabilities of VPN and IPv6, etc., couldhowever easily be violated. Many malware applications introduce and build existing network clients on a day-to-day basis as Botnet members. It has also been identified that the process to identify the botnet was based either on the IPv4 connectivity rather than on IPv6, and again the solutions implemented were based solely on almost obsolete anomaly detection techniques, and the malware is more dynamic than a signature-based model. The following are so few of the points in the legacy VPN:

1. The botnets or even the Clients were the only passwords protected and it’s easy for hackers to hijack the passwords.
2. The client’s devices were unidentified and untraced.
3. New users and devices were insecure.
4. Data transmission is unsecured between users to VPN servers.
5. Unwanted requests are more than legitimate clients.
6. Once the user connects to a VPN then they have all the rights of the LAN and all otherresources.
7. There is no centralized logs Server.
8. Only manual configuration for Firewall, IDS, and IPS policies.
9. Improper log analysis.

10. No Machine learning algorithms are implemented to train the dataset.

11. Manual updating of Repositories and databases.

12. Very difficult to synchronization and integration of Application services and API’s

13. DHCPv6 server does not support MAC Binding.

14. By default, the MAC address does not embed in IPv6.

15. By default, IPsec does not implement.

The industry, government organizations, enforcement agencies corporate could certainlybenefit from this research to run their IT infrastructure under the Secure Shield through the proposed framework. So a comprehensive solution that takes all of the above points into account and therefore becomes a strong mechanism for mitigation is needed. Consequently, it has been proposed a risk assessment and mitigation framework for Botnets on VPN's IPv6.

CHAPTER 3: RESEARCH METHODOLOGY

This section is followed by the justification for choosing a specific investigative methodology.

3.1 Introduction: Research Approach

The focus of this chapter is on a specific methodology of research that describes a new methodfor research purposes. A research proposal follows a global methodology for drawing findingson the overall aim and includes experiments, surveys, models, and case studies. Several methods and quantitative and qualitative methods used for processing the whole research can be relevant within the research methodology. Thus, the overall methodology of research is the pragmatic approach to research (mixed methods). Therefore, pragmatic research allows the freeuse of any method, technique, and the process typically linked to quantitative or qualitative research. Each method has also been acknowledged to have its limitation and to complement different approaches. The method of analysis was used to obtain the data from the research sothat the problem formulation could be answered. Throughout this thesis and concentrate on practical information, a pragmatist research philosophy has been followed. The analysis of VPN Security, IPv6, SDN, and Botnet technology will take both subjective and objective pointsof view into consideration.

3.2 Types of Research Methods

Research approaches include research plans and procedures ranging from broad assumptions to detailed methods for collecting, analyzing, and interpreting data. (Research, 2020) Three research approaches follow:

Qualitative

Quantitative, and

Pragmatic (Mixed methods)

Qualitative research is an approach to explore and understand the social or human problems that people describe. The goal is not to give a large population sample surface description, butto gain a profound understanding of one specific organization or event. This type of research focuses more on how people feel, think, and choose.

This study is largely conducted by discussion of concepts with certain open questions. Representatives are asked to explain why they have answered. This can demonstrate the underlying motivations, associations, and triggers of behavior ("Qualitative research",2020).

Common methods of collection are focus groups, thorough interviews, continuous observation,newsrooms, and the ethnographic participation/observation used in this research.

Quantitative research: The method of testing goal theories by examining the relationship between variables is quantitative research. This kind of research is a more logical approach, providing a statistical measurement of the mindset of the people. For example, perform the quantitative research, in case if required to know how many students use and how strongly theysupport Android Phones and its services.

This research is largely based on methods such as questionnaires and interviews that ask respondents to choose one or more of the options. Response options may include a scale for acceptance (to be strongly opposed), a scale for Likert, priority ranking, etc.

This can be done through phone, web, or paper questionnaires. This type of research is possible.The only limitation is that the number of participants should be sufficiently large to produce directional results.

Following Table 3.1 illustrates key differences between qualitative and quantitative research

Criteria	Qualitative	Quantitative
Data	Data is in word, image, or object form.	The figures and statistics arein the form of data.
Method used	Methods include focus groups, detailed interviews, and document reviews for non-numeric data.	Surveys, structured interviews and observations,and reviews of numerical information records or documents
Process of Research	Inductive approach for formulating theory or assumptions	Deductive approach for testing concepts, constructions, and hypotheses specified beforehand.
Response options	Options for unstructured or semi-structured response	Options for Fixed Appropriate response
Statistical test	There are no statistical tests.	For analysis, statistical tests are carried out
Time required	The time spent during the planning phase is less than the analytical stage.	The amount of time spent on planning is greater than during the analysis phase.
Objectivity/subjectivity	Highly subjective	Highly objective
Result	The results depend on the researcher's skill and precision	Results depend on the instrument/device
Final Report	The report contains textual information and textual details from participants in the research	The report includes statistical data analysis.

Table 3.1: Qualitative versus quantitative research

3.3 Method of Data Collection

The data gathering method was based on parameters such as the sample size, confidence level,and the percentage margin of error. Increased or decreased value correlation is listed below:

Parameters:	**Value increased**	**Value decreased**
Margin of error	Accuracy decreases	Accuracy increases
Confidence level	Accuracy increases	Accuracy decreases
Sample size	Accuracy increases	Accuracy decreases
Population size	Accuracy decreases	Accuracy increases

Table 3.2: List of Parameters

Calculating Sample Size: The following formula (Stokes, 2017) is used to calculate thesample size required.

Formula:

Necessary Sample Size = (Z-score)2 * StdDev*(1-StdDev)/(margin of error)2

Desired confidence level	**z-score**
80%	1.28
85%	1.44
90%	1.65
95%	1.96
99%	2.58

Where z-scores for the most common confidence levels:

Necessary Sample Size= $\frac{1.96 \text{ X } 2 \text{ X } 0.5 \text{ X } (1\text{-}0.5)}{0.05 \text{ X } 0.05}$ = 384.6 = **385**

Our sample size= 32 (interview respondents) + 363 (survey respondents) =395

The list of procedures used for Data Collection are as follows:

The technique of semi-structured interviews: It is used for collection of data. The chosen respondents were from the CCNA Group as well as members of the online troubleshootinggroup from companies, data centers, and IT infrastructure providers. Depending upon the area of interest, the network groups are mostly connected. Furthermore, conducted numerous webinars, conferences, online and offline training, etc. due to the subject. As a speaker at conferences at various times (both international and national) discussed ongoingresearch on common challenges in implementing VPN security, IPSec, IPv6, End-to-Endencryption, etc. As a participant in other conferences the opportunity in all these lectures to communicate with Cyber Security practitioners, auditors, CTOs, and regulators to understand the real-time problems and challenges in implementing and maintaining VPN. Especially during Covid-19, there was an enormous demand for IT professionals who could implement the VPN for organizations, to enable employees to work at home. The another way opted to interact with my Linked-in friends. The discussion held on the Relevant topic of this research and interacted them over the telephone and asked for a fewsample sheets/templates in the organization. For interviewees, the collected sample size was 32.

The questions for the interview were usually informal. The implementation and types of security challenges facing them, however, were entirely technical and more related to pros andcons. Furthermore, it was also concentrated on how these complaint tickets are handled or to understand their tactics.

There were less formal questions. More open questions have supported me in evaluating the current security scenario in VPN and I become more confident that my research in the same area will take place during interviews. During the discussions, it was able to gain a strong insight into VPN security risks and even new implementation, that the cost of applying securitysolutions and the problems and how they handle current software and hardware systems for troubleshooting is the way they address those risks when it comes to implementing and maintaining the VPN infrastructure. Questionnaires also dealt with actual threats to the existingsystem and contingencies in the event of any hacking event, as well as discussing whether or not they have a disaster recovery plan.

During interviews, the following list of sample questionswas posed. The answer to the selected questions is also in the majority as follows:

For references find attached Annexure - 3.1 carries Survey Questions Used for Quantitative analysis.

Surveys: The survey was performed with Google formats and printed hardcopy formats. Out of more than 500 requests, a total of authentic 363 respondents completed this form. The majority of respondents had excellent IT backgrounds and also include top IT infrastructure or management experts in the network, DevOps, and IT infrastructure industries. Few of them are the deciders for their organizations, where few of them work solely in the application and resolution of problems related to these areas. They belong to technical/non-technical organizations, but they are using or providing security or VPN implementation, etc. The survey was conducted between August 2019 and December 2020.

In-depth interviewing with individuals: or a focused set of interviewed participants wasused. This technique was employed because it helps the respondent to explore a specificidea, program, or solution from a detailed perspective. When its required to explore new problems and ideas in-depth, it is very useful. The laddering questioning technique was used when one issue led to a different question. This technique is especially usefulin the early stages of research as it gives the researcher direction.

These types of interviews and surveys have greatly supported the completion and collection of

data. There were about 15-20 minutes of each interview. It would be a separate interview or afocused interview with a group. The sample size was about 30-35 people and sampling techniques were used when sampling participants were asked to identify other potentials, and the chain continued until the right sample size was found. Snowball sampling is especially helpful when the researcher cannot find the number of participants needed and is also suspectedof disclosure of their identities in the case of potential participants. Some security workers do not want to interact because of the NDA that they signed or even are not confident in sharing the details of their organization's infrastructure, because of the faith in the trust or fear factor that old technologies might still be employed.

3.4 Research Methodology used

The study covered all aspects of essential methods of research used at various stages in this research topic. The research was investigative until the problem statement had been identified and "WHAT is the problem" was concentrated. The other phase was descriptive and did not identify and clarify the description of the "why of the problem" The main time consumption was experimental, in which several experiments were carried out and so that the correct "SOLUTION of the problem" could be found. The study thus encompasses all aspects of research. The effort placed to collect information on the problem the infrastructure industry is facing today. In-depth interview technologies collect information. Many experiments have been undertaken to produce the facts to provide the solutions for the framework proposed. In addition, many tests on various parameters were conducted to identify the impact of the proposed solutions. The start-up companies were aimed, once the results were good, for alphatesting, to provide them with cheap, free and secure solutions and to find real-time scenarios and the impact of the solutions proposed. Research began with a quantitative approach to quantify the different parameters of the VPN configuration used in different sectors of the organization. In the existing infrastructures, it also helped to identify the common problems, challenges, and vulnerabilities The qualitative research approach to solve the different common problems, issues, limitations, and safety parameters are further adopted. The process was fullyexperimental and required solutions that were results-oriented or customized, different in the different organizational sectors. Overall, the results of the hypothesis have been re-quantified,and because of the future approach, it is necessary to offer the latest innovation in the resource-sharing model with the VPN. Overall, the mixed approach to research for this research was more efficient and suitable

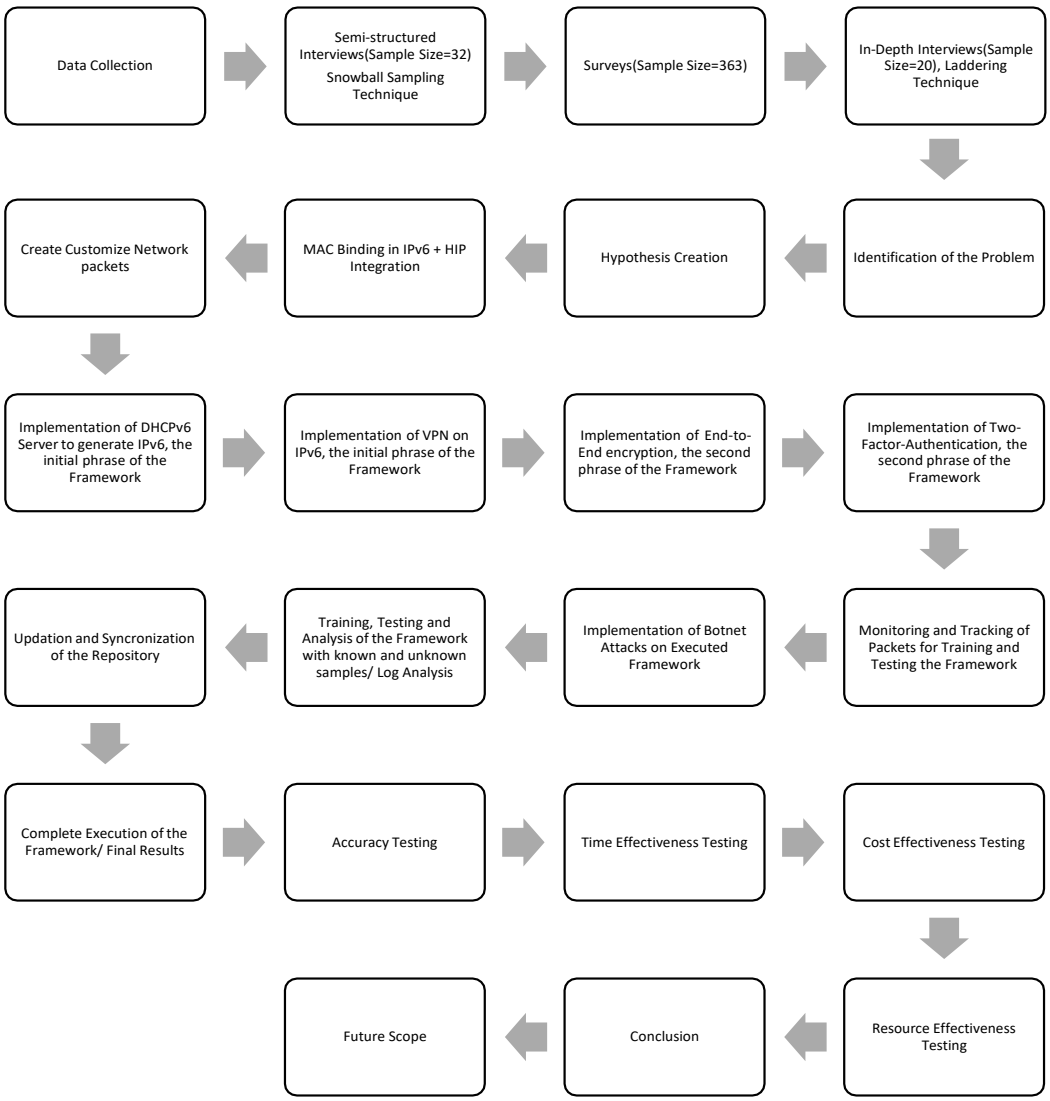


Figure 3.1: Research Methodology used

3.5 Survey Questionnaires

The survey questions are split into various terms to identify problems in real-time. The following are:

1. Professional and Contact Details
2. Organizational Details
3. Technical and Infrastructure Details Organizational Strength and Opportunities
4. Organizational Limitations (Management and Technical Side) etc.

5. For references find attached Annexure - 3.1 carries Survey Questions Used for Quantitative analysis.

3.5.1 Analytical Report of Collected Data

The following table shows the location and designation of interviewees and surveys conducted for the organizations/companies covered:

S.No	Name of Organization	Location	Designation of respondents
1.	Press club of India.	New Delhi, Mumbai	Cyber Security Consultant
2.	Softech Computers Delhi Ltd.	New Delhi	Manager Information Security
3.	GMMCO Ltd.	Chennai	IT Security Head
4.	AVTEC Ltd.	Bengaluru	Internal Auditor
5.	HDFC Bank	New Delhi	Chief Information Security Officer
6.	GAIL Ltd.	New Delhi	IT Manager
7.	Shoppers stop	Mumbai	Senior Manager, IT
8.	HIL Ltd.	Andhra Pradesh	Director
9.	Sedulity Solutions and Technologies	New Delhi	CEO
10.	Supreme Court of India	New Delhi	Advocate and Cyber Evangelist
11.	Reseller Club	Mumbai	IT Security Head
12.	Birla Soft pvt. Ltd.	Bengaluru	CIO
13.	Westside Ltd.	New Delhi	CEO
14.	Mindtree pvt. Ltd.	Bengaluru	Manager Information Security
15.	Rolta pvt. Ltd.	Mumbai	Information Security Analyst
16.	Infosys Ltd.	Bengaluru	IT Manager
17.	Bharti Airtel Limited	New Delhi	IT Audit Head

18.	Reliance Jio	Mumbai	Ex. IT Auditor
19.	Wipro	Bengaluru	Head - SAP Audits and IT Controls
20.	Jammu University	Jammu	IT Head, Networks
21.	Fintech Pvt. Ltd.	Gurugram	Head Operations
22.	Source Fuge pvt. Ltd.	Noida	Internal Auditor
23.	Tech Mahindra	Pune	Senior Manager, Information Security
24.	Cognizant	Hyderabad	Internal Auditor
25.	Goldman Sachs	Bengaluru	Internal Auditor
26.	Mphasis	Bengaluru	IT Head
27.	GE Capital Ltd.	Gurugram	Director
28.	Kerala Cyber Cell.	Kerala	Information Security Analyst
29.	3s Export House.	Noida	Cyber Security Consultant
30.	C.K. Birla Hospitals Enterprises Ltd.	Gurugram	Head Operations
31.	ITC Welcome Group Ltd.	Gurugram	Software Engineer
32.	Canara Bank	Bengaluru	Chief Information Security Officer

Table 3.3: List of Organization where survey Conducted

3.5.2 Location of organizations

Among all the respondents of the interview and survey conducted, 32% of organizations are set up in Delhi. 28% and 20% of organizations are set up in Bengaluru and Mumbai respectively. 11% of organizations are in Pune and the remaining 6% and 3% are in Gurugram and at other locations respectively.

Location	Count
New Delhi	125
Bengaluru	110
Mumbai	80
Gurugram	24
Pune	45
Other locations	11
Total	395

Table 3.4: list of location where survey conducted

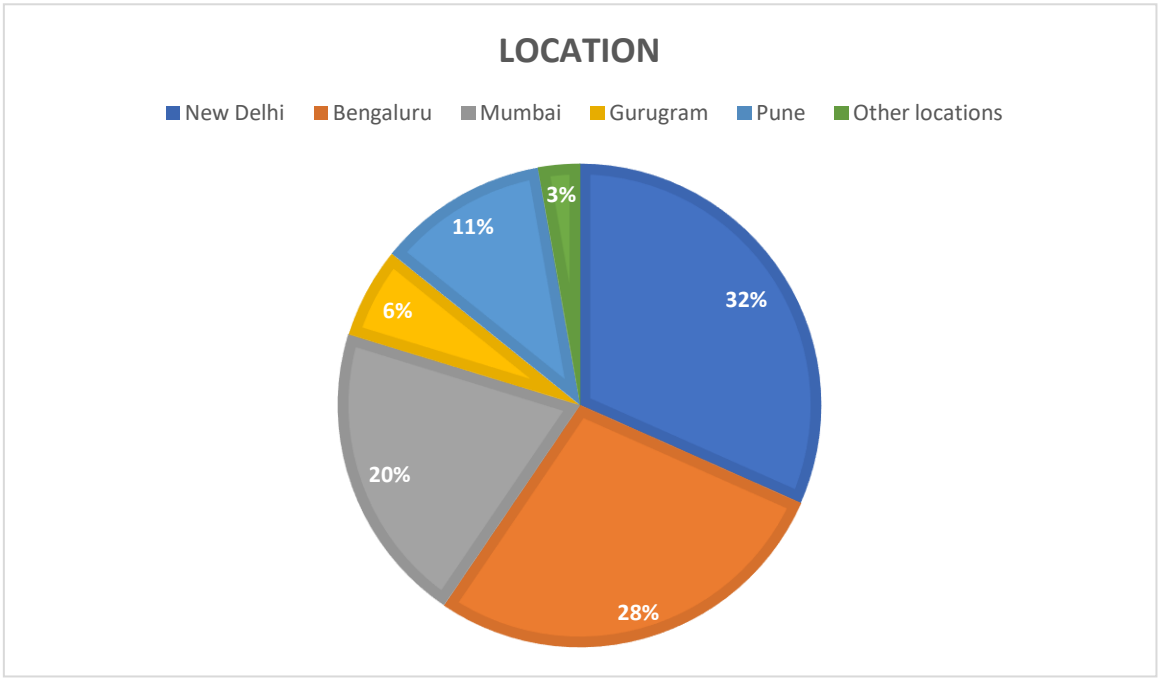


Figure 3.2: Graph-based on location

3.5.3 Type of Organization

Among all the respondents, the majorly targeted organizations are Telecommunications,Information Technology, and Financial/Banking services. These organizations are more vulnerable to cyber-attacks. Thus these organizations form 77% of the total organizations takenin the survey.

Type of Organization	Count
Information Technology	9
Telecommunication	9
Financial/Banking	6
Healthcare/Medical	3
Education	2
Government	1
Retail	2
Total	32

Table 3.5: Type of Organization

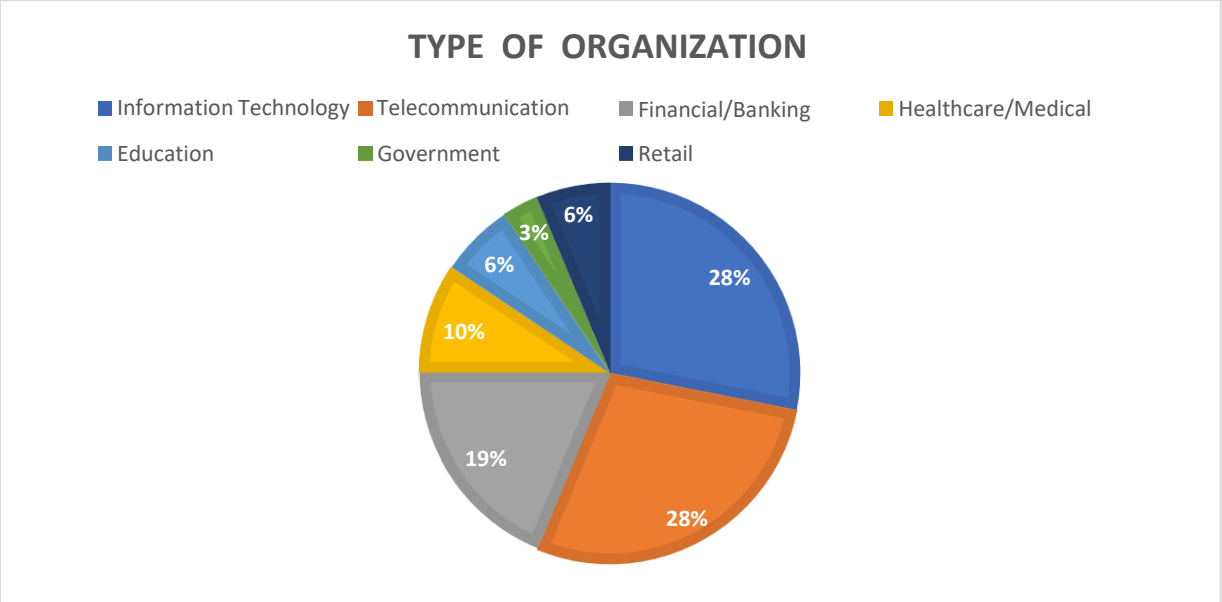


Figure 3.3: Graph of type of organization

3.5.4 Degree of Responsibility towards IT Infrastructure/ Data Security in theOrganization

Out of all the respondents, 51% were primarily responsible for security in their organizations.41% held secondary responsibility while only 8% were those who were not always responsiblefor security but only when needed.

Degree of Responsibility	Count
Primary Responsibility	200
Secondary Responsibility	162
Responsible when required	33
No Responsibility	0
Total	395

Table 3.6: Degree of responsibility

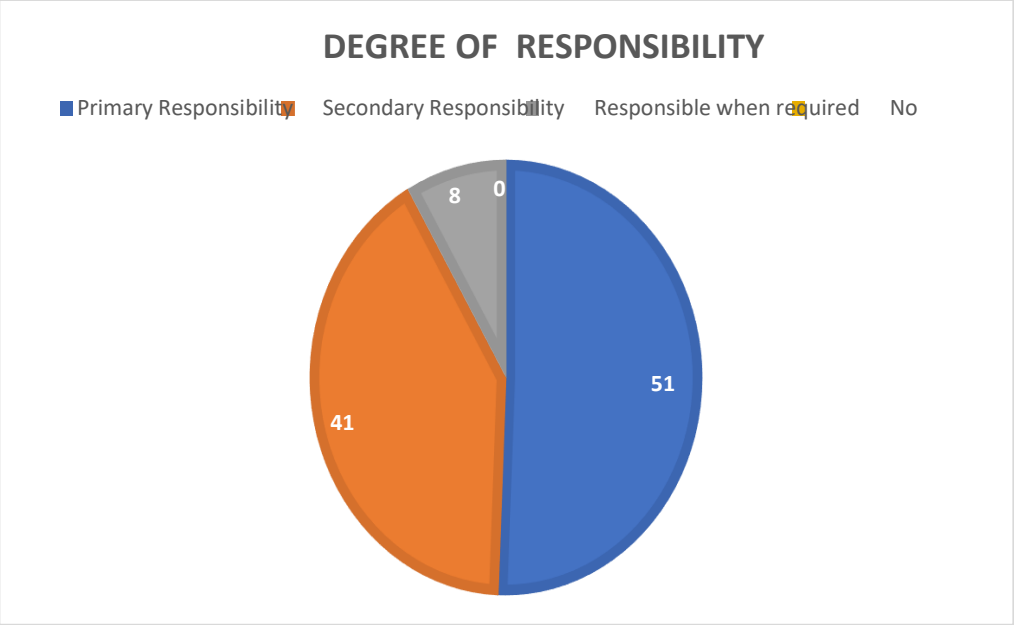


Figure 3.4: Graph of Degree of responsibility

3.5.5 Remote User Authentication allowed in the organizational network

Among all the respondents, 75% of them have an allowance for Remote User Authentication in their organizational network. 19% of them do not have the allowance to it and the remaining6% are given allowance on-demand or when needed.

Allowance	Count
Yes	295
No	75
On demand	25
Total	395

Table 3.7: Remote users allowed

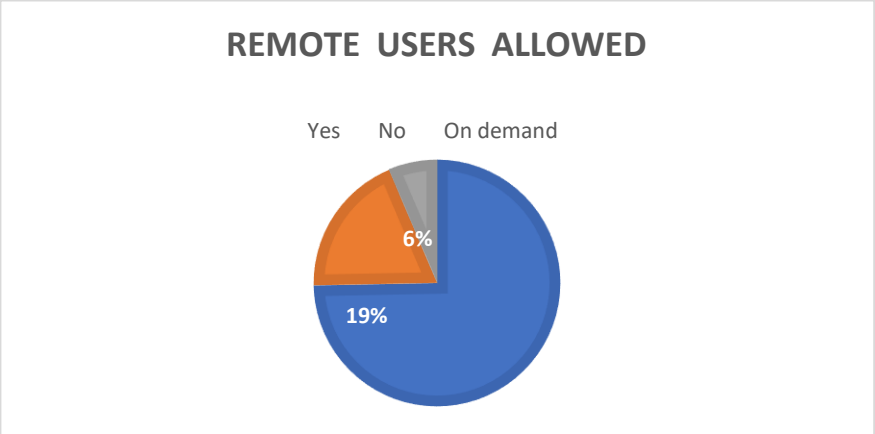


Figure 3.5: graph of Remote users allowed

3.5.6 IPv6 Implemented in the organizational network

Among all the organizations of the respondents, 78% of organizations have IPv6 implementedin their network. Whereas, 11% of organizations have not implemented IPv6 or implement it on-demand/when needed.

Implementation	Count
Yes	42
No	313
On demand	45
Total	**395**

Table 3.8: IPv6 Implemented

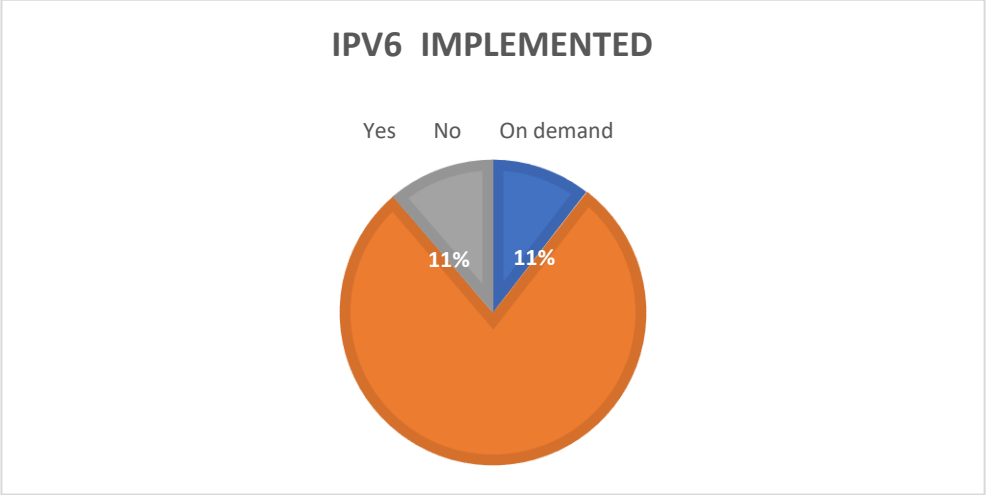


Figure 3.6: Graph of IPv6 Implemented

3.5.7 VPN implemented in an organizational network

Among all the organizations, 67% of organizations have implemented VPN. Whereas, 23%and 10% of people have do not implemented or implement on-demand respectively.

Implementation	Count
Yes	264
No	91
On demand	40
Total	**395**

Table 3.9: VPN Implemented

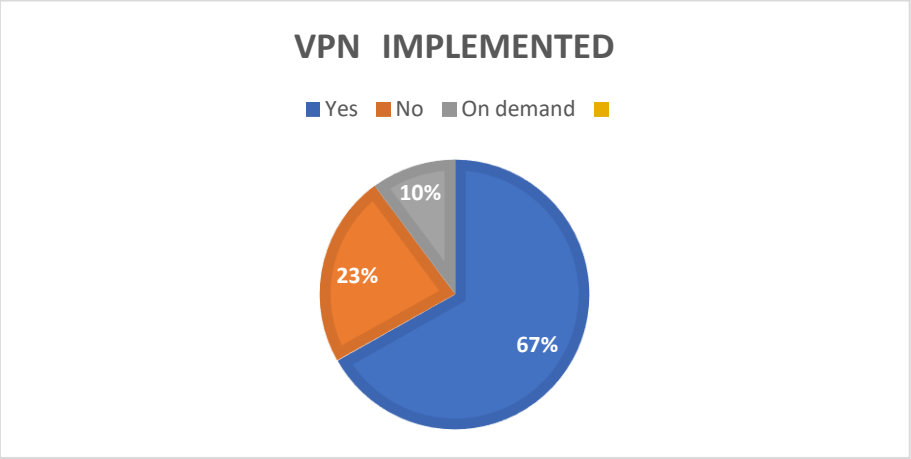


Figure 3.7: Graph of VPN Implemented

3.5.8 Maintain End User Trusted devices in the organizational network

Among all the organizations, 59%of them maintain end-user trusted devices in their network.39% of them do not maintain end-user trusted devices and 2% of them do it on demand.

Maintain Trusteddevices	**Count**
Yes	154
No	233
On demand	08
Total	**395**

Table 3.10: Maintain trusted devices

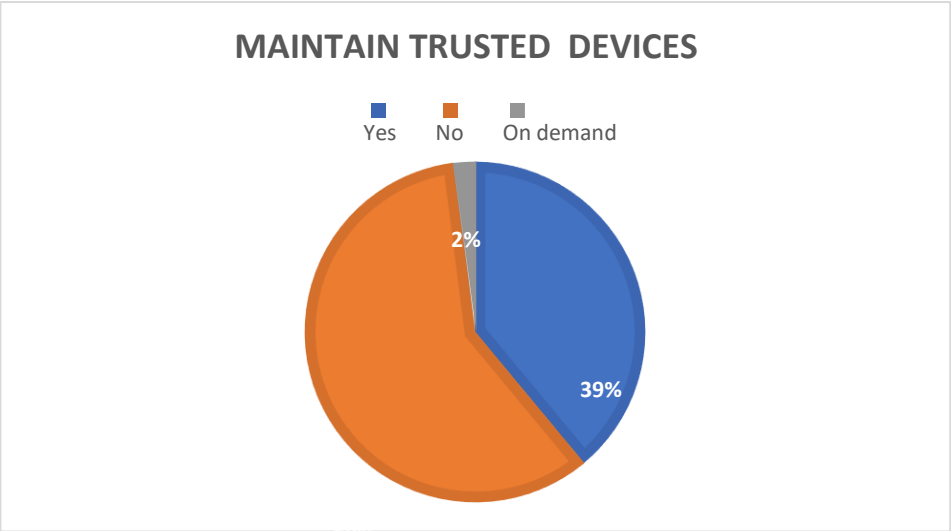


Figure 3.8: graph of Maintain trusted devices

3.5.9 Hardware Firewall in the organizational network

Among all the organizations, 70% of them have Hardware Firewall in their network. Whereas,26% of them do not have a hardware firewall, and 4%of them can arrange if required or demanded.

Firewall	**Count**
Yes	276
No	102
On demand	17
Total	**395**

Table 3.11: Hardware Firewall in Organization

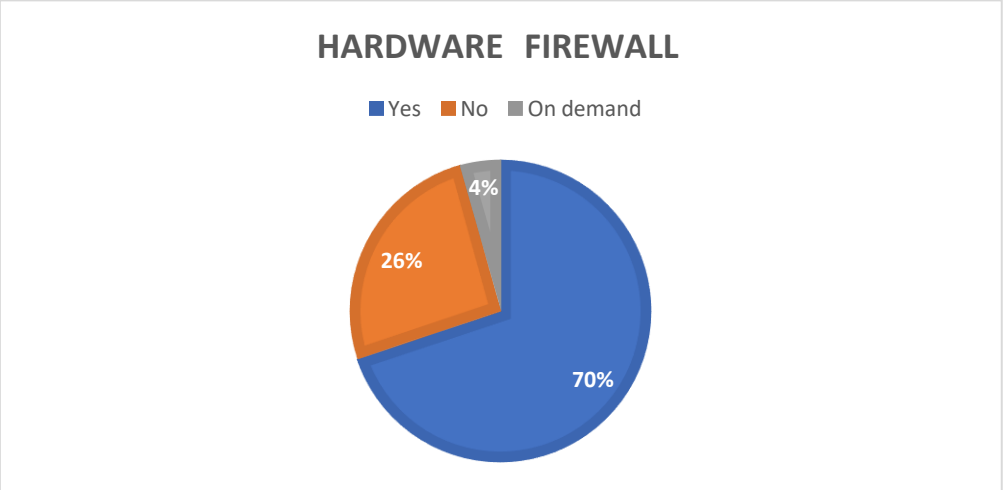


Figure 3.9: Graph of Hardware Firewall in Organization

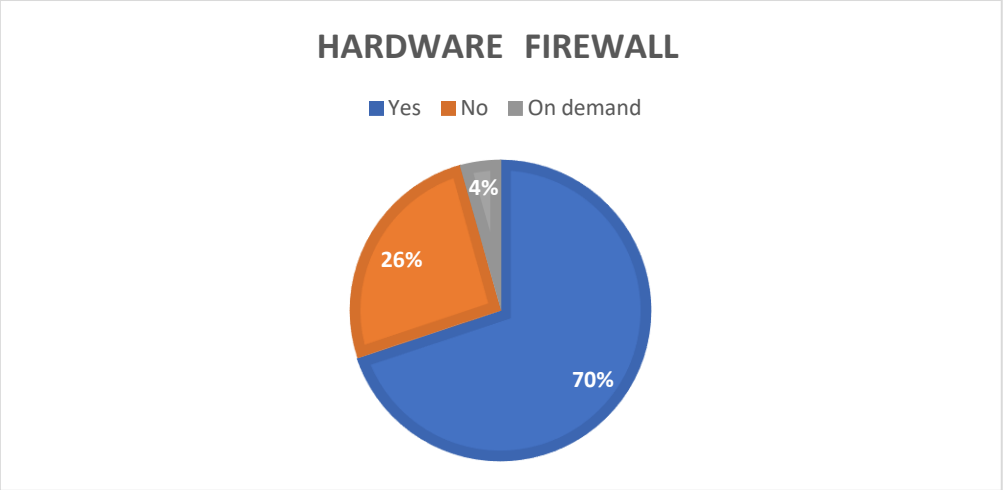


Figure 3.9: Graph of Hardware Firewall in Organization

3.5.10 Cyber Security budget

Most of the organizations' cybersecurity budget is 10-20% of the total budget. Other organizations spend more or less than 10-20% according to their technical needs.

Percentage	Count
<10%	78
10%-20%	142
20%-30%	72

30%-40%	51
40%-50%	39
>50%	13
Total	**395**

Table 3.12: Cyber Security Budget

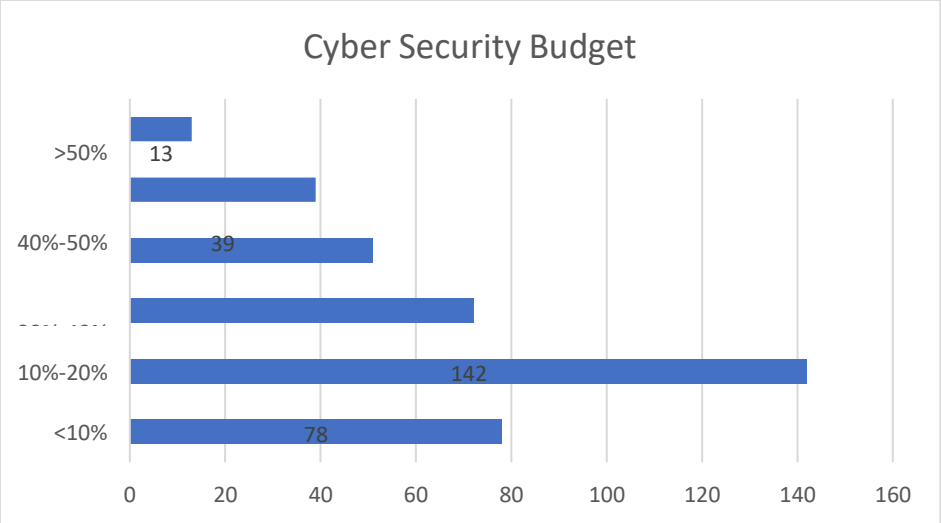


Figure 3.10: Graph of Cyber Security Budget

3.5.11 Type of Devices/ Platform connected to organizational Network

Among all the organizations, the given bar graph shows that the interest in the use of differentoperating systems is diversifying. Windows has the highest users among all the respondents and their organizations. Although, other OS like Linux, android, mac os are also used.

Platform type	**Count**
Windows	114
Linux	79
Apple OS	56
Android	60

All the above	86
Total	**395**

Table 3.13: Platform type

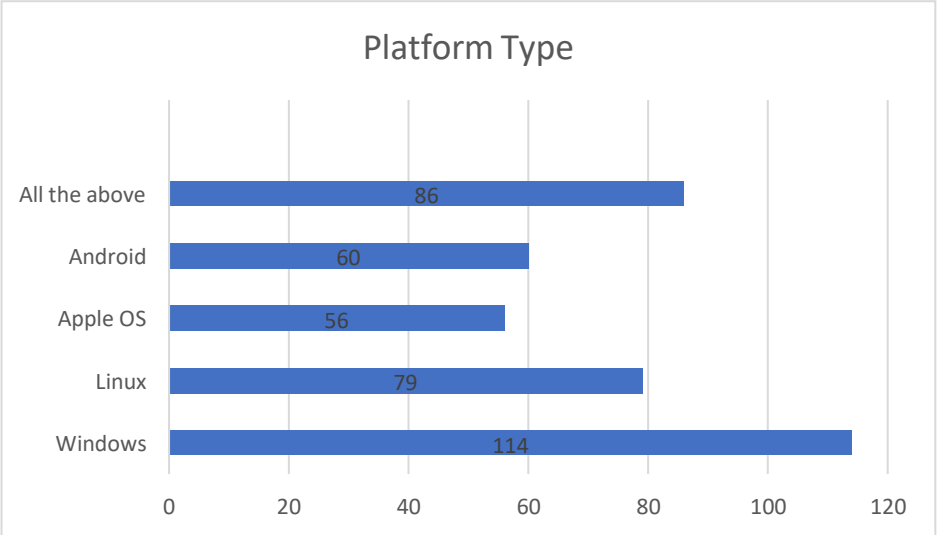


Figure 3.11: Graph of Platform type

3.5.12 The level of Cyber Security in the organizational network

Among all the respondents, the moderate level of cybersecurity is seen in more organizations than high or low levels. The following bar graph clearly shows the high, moderate, and low levels of cybersecurity together with not concerned organizations.

Level	**Count**
High	106
Moderate	158
Low	122
No concern	09
Total	**395**

Table 3.14: Cyber Security Level in an organization

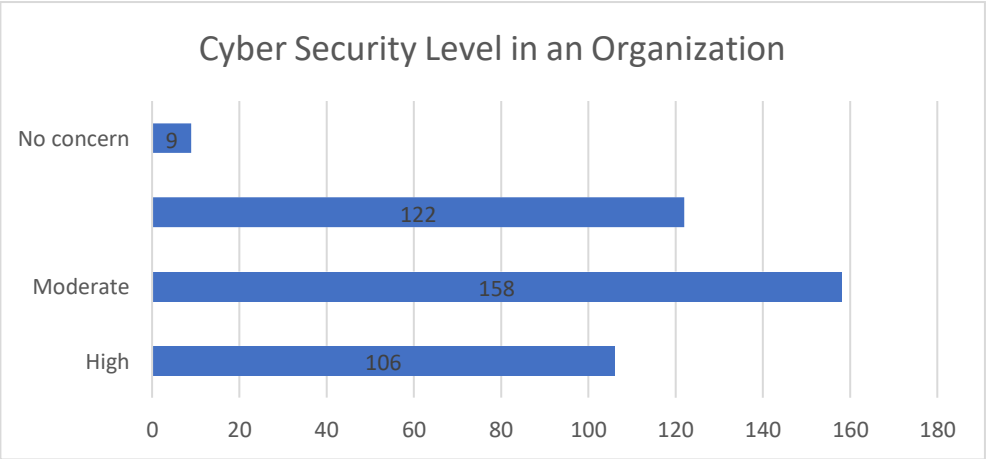


Figure 3.12: Graph of Cyber Security Level in an organization

3.5.13 The approach to identifying the legitimate devices connected to the Network

The following bar graph displays the approaches used by respondents to identify the legitimate devices that are connected to the organizational network. The most used approach is to identifyby IP address of the devices. Whereas, other approaches like Firewall, Mac address, log server,and user credentials are also seen.

Approach	**Count**
IP Address	142
Firewall	90
Mac Address	78
Log Server	39
User Credentials	46
Total	**395**

Table 3.15: Legitimate devices identification

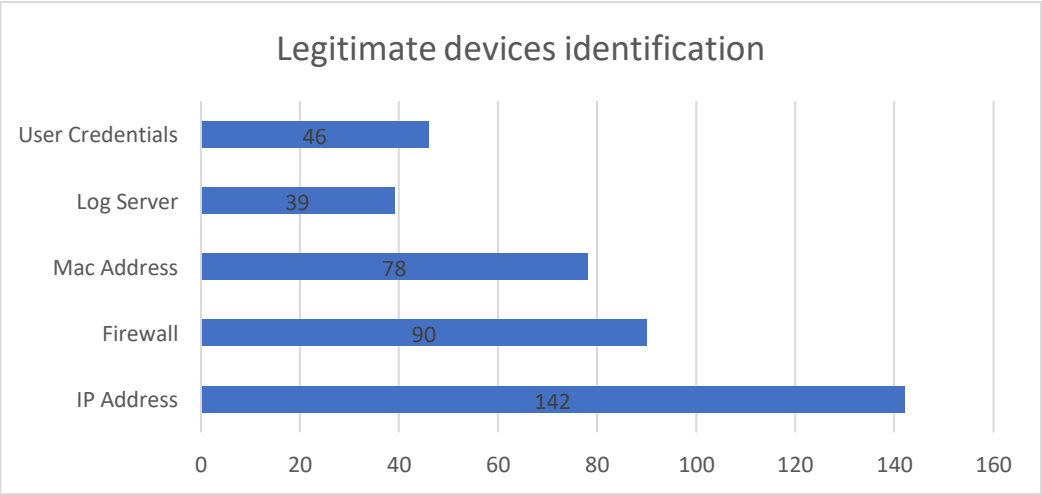


Figure 3.13: Graph of Legitimate devices identification

3.5.14 The average response time for Incident Handling

The given bar graph displays the average response time taken by organizations for incident handling. Most of the responses show that their organizations take more than a day or a day forincident handling which is followed by less than an hour, 1 week, and more than a week.

Average Time	**Count**
<1 hour	35
1 day	118
>1 day	132
1 week	70
>1 week	40
Total	**395**

Table 3.16: Response time for Incident Handling

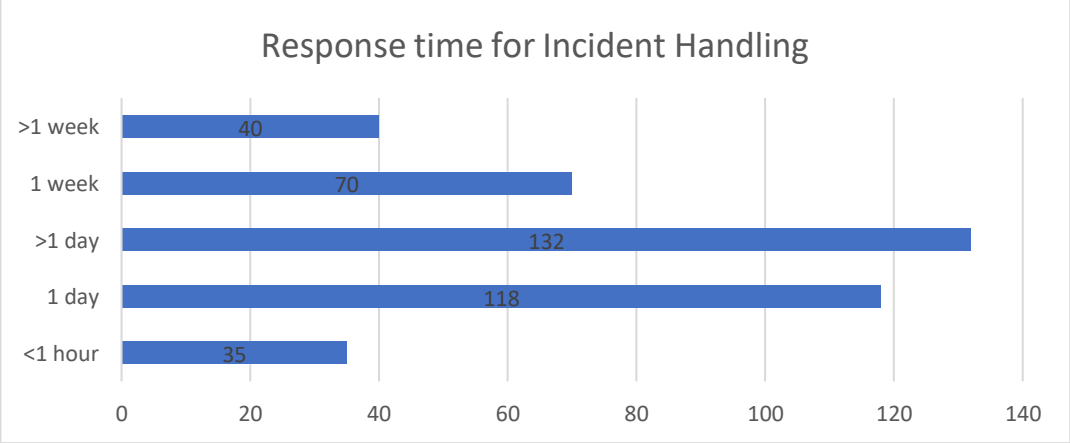


Figure 3.14: Graph of response time for Incident Handling

3.5.15 The important factors of Cyber Security Challenges in the organization

Among all the respondents, most of the organizations face Malware attacks and Intranet Security as the major challenges of cybersecurity which are henceforth, followed by user authentication and granting privileges, endpoint security, and miscellaneous threats as displayed by the given graph.

Cyber Security Challenges	**Count**
User Authentication and Granting Privileges	39
Malware Attacks	134
Intranet Security	118
End Point Security	49
Miscellaneous Threats	55
Total	**395**

Table 3.17: Cyber Security Challenges

3.6 Key Understandings

The key understandings after an analysis of the collected data are as follows:

1) Organizations allocate a very less percentage of overall budgets to cybersecurity.

2) Organizations face lots of malware attacks on existing VPNs.

3) Security persons place an entire load on firewalls and IDPs to control maliciousrequests, but once the permission is granted then the user has full control.

4) Lack of regular training amongst employees that how to use the configuredinfrastructure, and in case of any incident than how to respond.

5) There is no way to recognize end-user devices and their path.

6) Attackers are using advanced AI and ML techniques for the creation of malwarewhereas the current security mechanisms are unable to handle them.

7) The average response time of the security team is very high which leads to anincident response in the entire organization.

8) The level of concern for cybersecurity in organizations is low to moderate.

9) Large enterprises are using VPN as a Service in the corporate level HardwareFirewall/ Routers which are very expensive and vulnerable too.

10) It also reduces the performance of the network speed and bandwidth.

3.7 Experimental Identification of the Problem

3.7.1 Creation of Hypothesis

Many experiments that are categorized in various forms to implement the proposed frameworkare required.

1. Preliminary Settings
2. Implementation of existing Portfolio to initiate Experiments.
3. Implement Research Gaps
4. Implementation of the System Training

5. Testing with Bots
6. Results and Analysis
7. Facts and Findings.

3.7.2 Hypothesis of Study

VPN is one of the hackers' focused networks. Organizations, researchers, and solution architects have always attempted, through various means, to identify their safety measures related to VPN implementation.

It was decided to carry out several hypothetical experiments such as the Complex Hypothesis,the Directional Hypothesis, and the Null Hypothesis to identify real-time and understand the ideology behind the implementation of various alternatives and processes. The focus is on generating experimental results. These experiments help identify gaps and limitations in research and to analyze the results that have been demonstrated.

3.7.3 Testing of Hypothesis

3.7.3.1 H01 - Case Study 1.

Titled: Problems with VPN connection during IPv6 connection.

Introduction: IPv6 addresses and headers occupy more space than IPv4 addresses and headersin the data packet. Due to this, some users are not capable of connecting to the VPN and otherscan log in, but they cannot download files or read emails, or do other items in the data packetsthat use large data payloads. By adjusting the MTU packet to a lower value, there is sufficientspace for the larger IPv6 headers in the packet.

This only applies to clients connecting via IPv6. AnyConnect software from Cisco will alwaysuse IPv4 when available, which mainly impacts on customers who use open connect or clientswith IPv6 only (which is rare). The MTU standard is 1500 bytes for wireless and Ethernet. In

IPv6 there is a need to set it to between 1380 and 1450 MTU, depending on configuration, particularly if it is tunneled.

Objective: Connection problems when connecting via IPv6 to identify the VPN. Based on theidentified problems, it was proposed that the risk of discontinuity or unavailability of the connection be mitigated with few settings.

Implementation steps to mitigate the Issues in VPN connectivity with VPN on IPv6.

- If a connecting client is not available and the message appears i.e. "time out or trying to connect" (and uses IPv6 to get to VPN*), use the ping6 command on VPN (Unix/MacOS is the command "ping6 VPN SERVER DOMAIN/IP"). This isn't the problem if it doesn't work.
- If ping6 is working, try accessing the website via IPv6 (or any of the VPNs). Probably this is not the problem if it loads. The problem could be if the VPN is loaded and connected, but certain items don't work.
- If ping6 worked but the website wasn't loaded, the appropriate problem is likely to arise. Try modifying the MTU configuration and see if the problem is corrected.
- It is required to set the MTU size to a minimum of 1450, if the problem identified, possibly just mention 1380 so that it works.

Mac OS:

The process of changing the MTU in the network configuration.

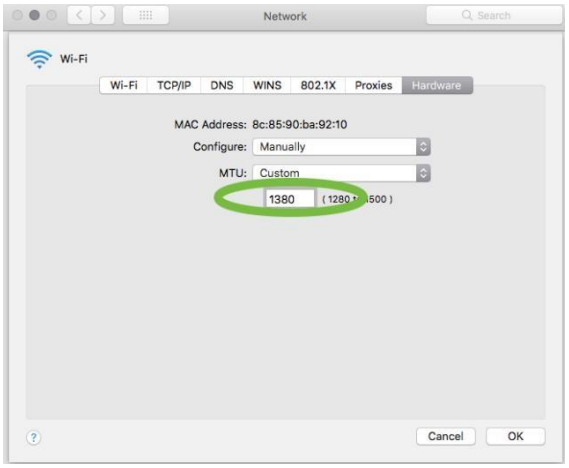


Figure 3.17: MTU settings in MAC OS

Windows OS:

Use the command prompt to pass the following command in Windows OS to set the MTUvalue.

```
netsh interface ipv6 show sub interface
```

Display the list of network interfaces.

```
netsh interface ipv6 set sub interface “Local Area Connection” mtu=1450 store=persistent
```

Repeat the steps above – replacing 1458 with 1430, or 1380 – if in case of any problems aftermodifying the MTU, restart the computer and test it again.

Linux:

There are many ways of doing this in Linux. Two options are available here:Use "-m"

to specify the MTU in this way to use openconnect.

```
openconnect -m 1380 -v VPN_SERVER_DOMAIN/ IPv6
```

If not, after connecting the VPN, set the MTU on the created tunnel interface (in this examplethe tunnel was tun0)

```
ifconfig tun0 mtu 1380
```

The process to identify that the client has been connected to the VPN over IPv6.

In case if the client is IPv4 only configured, which means that the client is IPv4 only, and the PC is not connected via IPv6. If IPv6 address is the only one found, which is the stat on IPv6.

In case if identified both an IPv4 and an IPv6 address and couldn't connect at all, it is difficultto specify which address required to link to the VPN. When using the Cisco AnyConnect software, IPv4 will always be utilized as a general rule, if it contains one. In case ofopenconnect, or any other free client, it probably uses IPv6 - IPv6 will be tried first by most open-source software. Probably, can check the logs for the connection if required to know andsee what IP address assigned in connection with.

If the connection is established but things don't work, the IP address of the connected server must be checked for the VPN application in such a case. It identifies whether IPv4 or IPv6 is involved. Open the window for statistics (on Mac click on the graph icon on the connection window, on Windows click on the gear icon on the connection window, then select the statisticstab). Then search for a server IP address line called

"Server." If the IPv6 address (up to 8 hex numbers separated by columns) is used, it can connect to the server via IPv6. Note: There may be fewer than eight colon numbers if a column doubles – such as 2620:0:e00:3a::2.

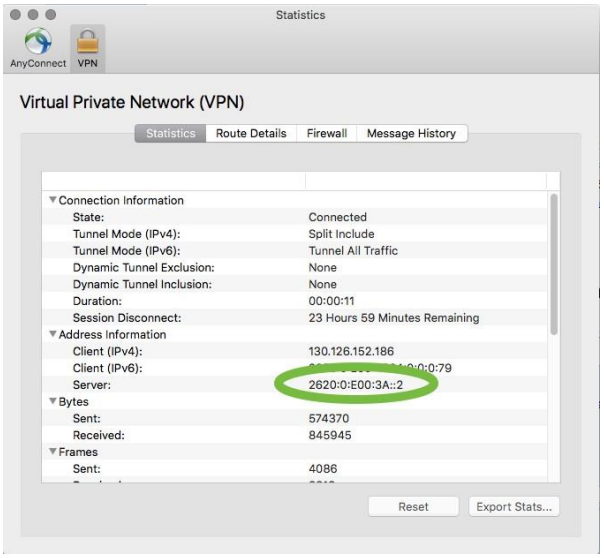


Figure 3.18: Identify Server IP Address

When the address is noted in IPv4 (4 decimal numerals separated by periods like 192.17.88.30),the connection does not seem to be IPv6.

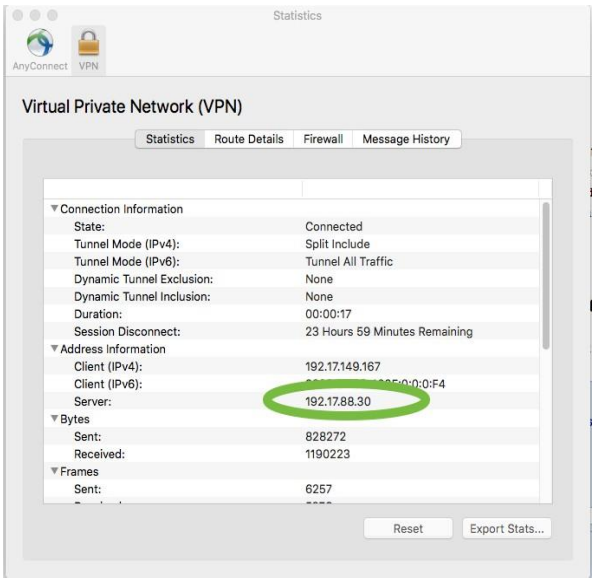


Figure 3.19: Identify Server IPv4

The procedure to prevent IPv6 VPN breakout

IPv6 traffic from remote devices can escape corporate security controls without properly configured remote-access VPNs.

Unconscious of the role played by IPv6 on remote devices, companies are likely to access banned websites even though VPNs are meant for restricting access to such machines.

This hole is because certain of the remote access VPNs only inspect and use IPv4 traffic security controls when they pass through a VPN concentrator, without allowing similar IPv6 traffic safeguards.

This allows IPv6 traffic to directly access the Internet without these checks. The problem iswell known but is often overlooked, known as an IPv6 VPN breakout.

Solutions for IPv6 VPN breaking out exist, but the first step is to grasp the importance

of it.The IPv6 VPN disintegration is ignored

Many companies fail to understand how frequently IPv6 is used on devices that use VPN to access their networks. Phones, tablets, and laptops are generally supported by IPv6, as are broadband and cellular services for access to the internet.

As a result, IPv6 is often not recognized as a security factor by companies. Set the VPNs up for traffic inspection only, which allows mobile devices to access IPv6 sites which can prove dangerous to commercial networks,

devices, and data. Once the VPN is established, the IPv4 protection system inspects traffic connected to the internet and blocks traffic for destinations assessed by the policies the company has set up. (see, Figure 3.20).

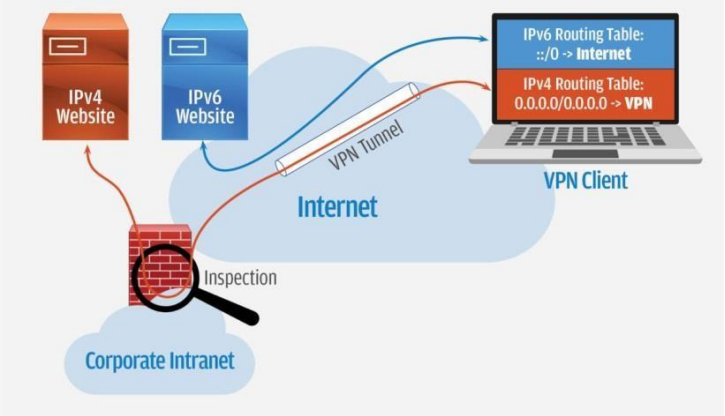


Figure 3.20: VPN connection

Network World / IDG

This diagram shows an IPv4-only VPN tunnel established on a typical VPN company laptop for the corporate Internet environment. The red line represents IPv4 traffic directed via the company to carry out traffic and security checks for internet traffic. All IPv4 traffic should betransported via a VPN tunnel and cannot access the Internet directly, but the blue-line IPv6 traffic is permitted.

Most company VPNs implement what is called split-tunneling, forcing all IPv4 connections topass through the VPN to improve safety. Without split-tunneling, remote devices cannot establish a separate Internet connection once they have established a VPN connection.

Conclusion: The steps and methods used above in VPN work with IPv6 to mitigate problems of connectivity. It has been tested on all the OS on the market, such as Windows, Mac OS, and Linux. Few options were shared to solve the connectivity challenges.

3.7.3.2 H02 - Case Study 2.

Title: Vulnerabilities exploited in VPN products used worldwide

Introduction: The NCSC (NCSC, 2019) is investigating the exploitation of known vulnerabilities affecting VPN products by suppliers Pulse Secure, Fortinet, and Palo Alto by Advanced Persistent Threat (APT).

This is an ongoing activity that is aimed at both UK and world organizations. Includes government, military, academic, commercial, and health care sectors affected. These are well documented open source vulnerabilities, and industry figures show that hundreds of UK hostscould be vulnerable.

Objective: The aim is to implement and identify the worldwide use of VPN products. In this scenario, it is required to discuss the exposed common vulnerabilities and the impact of the branded product attacks.

Several SSL VPN products have vulnerabilities that enable an attacker to recover arbitrary files, including authentication credentials.

To connect to the VPN and change configuration settings, or connect to other internal infrastructure, an attacker can use stolen login numbers.

An unauthorized VPN connection could also give the attacker the privileges required to perform secondary exploits for root shell access.

Sample Collection, compilation, and Implementation:

Top vulnerabilities: Although this is not an expansionary list of CVEs related to such products, the most common vulnerabilities known to be exploited by APTs are listed below.

Sample exploit code is publicly available online for these vulnerabilities. The NCSC warns against the untrusted third-party code testing infrastructure.

Pulse Connect Secure:

- CVE-2019-11510: Pre-auth arbitrary file reading

- CVE-2019-11539: Post-auth command injection

Fortinet:

- CVE-2018-13379: Pre-auth arbitrary file reading

- CVE-2018-13382: This allows an unauthenticated attacker to change the password of an SSL VPN web portal user.
- CVE-2018-13383: Post-auth heap overflow. This allows an attacker to gain a shell running on the router.

Palo Alto:

- CVE-2019-1579: Palo Alto Networks GlobalProtect Portal

Detecting exploitation: Users of these VPN products should examine their logs for compromise evidence, especially if patches may not be applied immediately after release.

In addition to the specific product advice below, managers should also seek proof of compromised accounting for active use, such as misplaced IPs or times.

There are open-source snort rules but cannot pick up events over HTTPS for exploitation.

Pulse Connect Secure: The best way to detect exploitation attempts is to find evidence of vulnerable URL connections on the device.
Press. Secure logging is highly configurable so that web applications can be checked for logging onto a system, an HTTPS application is made directly to the web interface, not via theVPN, and checked if the event logs show.

Vulnerability	Detection
CVE-2019-11510	Search logs for URLs containing **?** and ending with **/dana/html5acc/guacamole/** (Regular Expression: **\?.*dana/html5acc/guacamole/**) If a date is found before the application of the patch, a compromise may be shown. The filename the attackers attempted to read is given in the matching string.
CVE-2019-11539	Search for requests to **/dana-admin/diag/diag.cgi** with an options= parameter in the URL. An exploit will almost certainly contain: -r, # or 2> [Data between -r and # is Perl code that would be executed.]

Table 3.19: Vulnerability Detection

Fortigate: Fortigate devices do not log web requests on the standard basis, but if firewall logologs or NetFlow logs are available from another device, then it may be possible to detect exploitation if a device is configured to write logs for all connections.

An attacker can download the SSL VPN web session containing the active user usernames and passwords when using CVE-2018-13379. Typically this file is a minimum of 200 KB.

Firewalls, or NetFlow logs, may be retrieved from the operation proof for TCP sessions of 200,000 - 250,000 bytes from the web interface port of the SSL VPN device to the client, witha small number of bytes (with less than 2,000 of them) from the client.

Palo Alto: The security notice for a vulnerability (CVE-2019-1579) previously patched in August 2018 was issued by Palo Alto in July 2019. The versions below may be vulnerable:

- Palo Alto GlobalProtect SSL VPN 7.1.x < 7.1.19

- Palo Alto GlobalProtect SSL VPN 8.0.x < 8.0.12

- Palo Alto GlobalProtect SSL VPN 8.1.x < 8.1.3

In logs, past exploitation may be hard to detect. But unsuccessful exploit attempts can cause acrash that can be seen in logs.

Implementation: Arbitrary File Exploitation Secure Pulse SSL VPN Read (CVE-2019-11510)

One domain, either a domain list, can be used. Need to add https:// to the domain.

Usage :

```
cat targetlist.txt | bash CVE-2019-11510.sh / bash CVE-
2019-11510.sh –dhttps://SERVER_DOMAIN_NAME/IP
```

For a verification of the exploit, just download the file /etc/passwd then use:

```
cat targetlist.txt | bash CVE-2019-11510.sh --only-etc-passwd

bash CVE-2019-11510.sh -d https://SERVER_DOMAIN_NAME/IP --
```

The output will be saved inside output/ SERVER_DOMAIN_NAME/IP /Output :

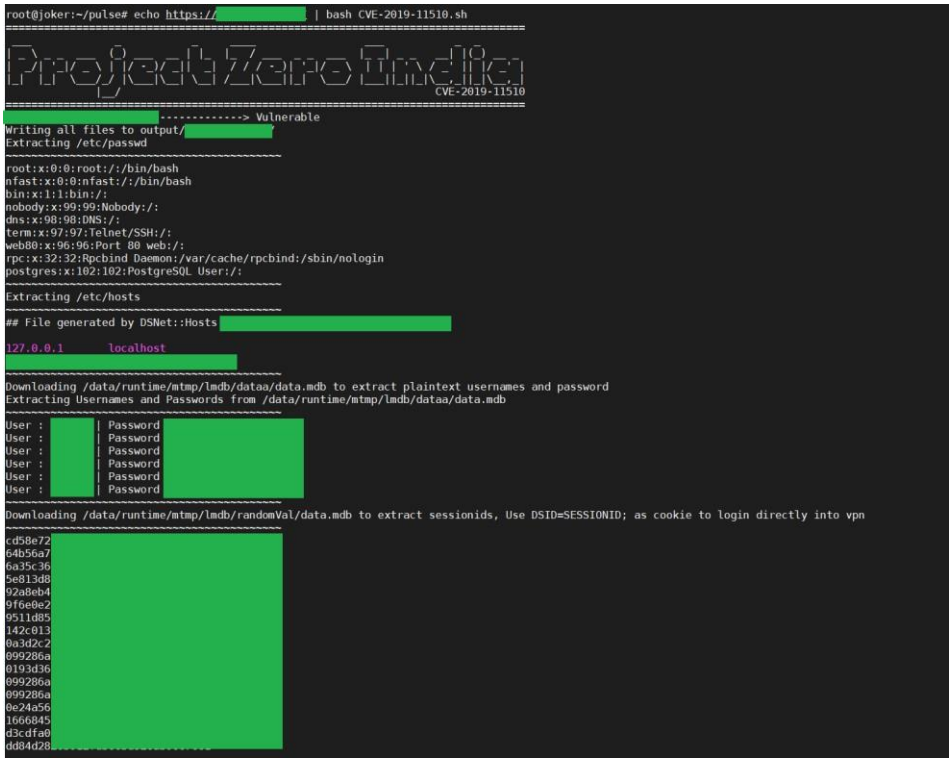


Figure 3.21: Output

Essential mitigation: Owners of vulnerable products should take two steps to mitigate thesevulnerabilities:

- Use the latest vendor-released security patches

- Reset the authentication of the VPNs and the connecting accounts affected

To mitigate the risk of the actors using these vulnerabilities, it is most effective to ensure that the product concerned is patched with the latest updates on security. All patches are available for these vulnerabilities in Pulse Secure, Fortinet, and Palo Alto. Security patches should always be implemented immediately.

The NCSC acknowledges that patches are not always easy and can cause disruption in some cases, but it remains the most important step for an organization or a person to take to protect themselves.

In case of suspected exploitation

System managers suspecting exploitation or not excluding that potential should revoke credentials at risk of theft. This can include administrative as well as user credentials.

The reset of authentication credentials will protect against unauthorized access by using credentials acquired before the affected systems were patched.

Conclusion: This experiment provided companies to audit their VPN devices and their configurations, which must be determined to mitigate the risk and to perform uninterrupted services at the same time.

3.7.3.3 H03 - Case Study 3.

Title: Mirai Botnet Affecting IoT DevicesVirus Type: Trojan/Backdoor

Severity: High

Introduction: Variants of "Mirai" (Cert, 2017) which aim at the Internet of Things[IoT] devices like printers, video cameras, routers, smart TVs have been found to spread. The malware can scan network devices or the website, in particular those that are protected with default login or hardcoded username passwords.

Objective: This Botnet has the purpose of affecting IoT devices that can connect to their network devices. In addition, it was another way of acting like a used VPN and offering the hackers commands and control services. The system used for DDOS attacks is also infected.

Sample Collection, compilation, and Implementation:

Compromise IoT systems using the default username and passwords: The malware can performthis function.

- Create compromise equipment botnets.
- To launch DDoS attacks, use compromise devices.
- Make network links to receive commands from launching additional attacks.

The malware is also reported to be in the memory of the infected device and can only be removed by rebooting the affected device. The malware nevertheless scans the vulnerable devices, leading to the rebooted device re-infection within minutes of rebooting.

The malware is also reported to be in the memory of the infected device and can only be removed by rebooting the affected device. The malware nevertheless scans the vulnerable devices, leading to the rebooted device re-infection within minutes of rebooting.

The new version of Mirai IoT DDoS malware "Satori"

"Satori," a new variant for the malware Mirai IoT DDoS, has recently been reported to spreadlike a worm. Satori, Mirai is not the new variant as it does not use a Telnet port Scanner, but scans 37215 and 52869 TCP ports on random IP addresses. It has two new exploits aimed at 37215 and 52869 TCP ports. One works with TCP port 52869 of IoT devices/routers that exploit [CVE-2014-8361] vulnerability within Realtek SDK's mini-sourced SOAP service.

In the samples of Satori 3 C2s are observed:

- 95. 211. 123. 69:7645
- network.bigbotpein.com:23
- control.almashosting.ru

Mirai Malware Package: The source code for Mirai malware has been published and writtenin C language. The major malware components are:

Call-home routine: This module makes network connections to the server for command and control. This module is initially executed on the compromised IoT device and connects to the server to receive information about attacks. **Set of attack routines:** The Network Traffic generates routines to disrupt the network capacityof victims.

Network Scanner routine: Scans the network of victims to discover other vulnerable IoT devices throughout the network and lists such devices for further damages and attacks.

Command and Control Tool: The malware uses the "CNC" control and control tool, writtenin "Go," to provide multiplatform support for 32-bit and 64--bit integer Chip, AMD, and MIPS chips for common home IoT devices, including seven different computer architectures. The malware is designed to run on both ordinary and hardware computers.

Under the following Mirai malware default IoT vulnerable username/password list is shown:

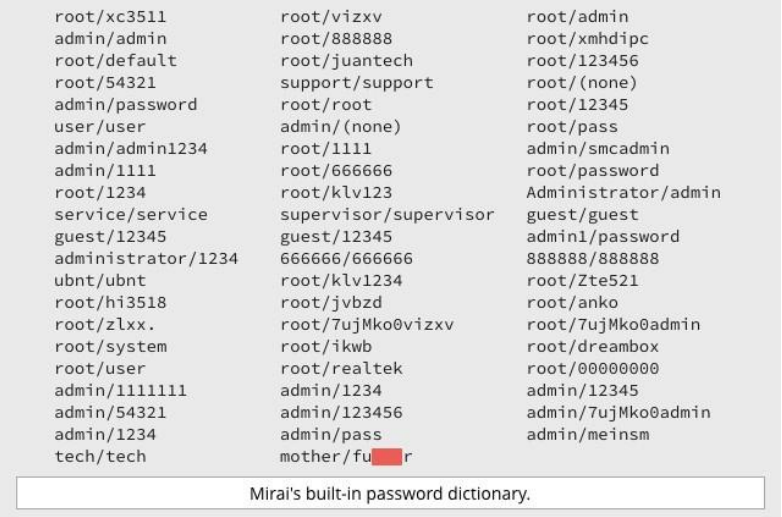


Figure 3.22: Mirai's built-in password dictionary

Indicators of compromise:

- Abnormal traffic to vulnerable devices on ports 2323/TCP and 23/TCP.
- Traffic on port 48101/TCP Command and Control Network.
- Riesling outbound traffic when the device is involved in the DDoS attack.
- Commitment indicators for "Satori":
- Port 37215/TCP and 52869/TCP traffic abnormal when scanning for vulnerable devices

- Traffic command and control over the network to these IP and domain addresses and domains: "95. 211. 123. 69:7645," "network.bigbotpein.com:23," "control.almashosting.ru"

Countermeasures and mitigation policies for securing IOT devices:

1. Restrict access to IoT devices by authorized users to Web Management Interface only and modify standard username/passwords
2. Change the default login credentials before production deployment.
3. Change the default credentials at the startup device and ensure that passwords are as complex as possible.
4. Disable UPnP on IoT devices, except where necessary.
5. Installed devices and their capacities should be made known to users. If a device has a default password or an open Wi-Fi connection, users are asked to change the password and only allow it to work with a secured Wi-Fi router on a home network.
6. Check the access to Access List devices
7. Configure "lock" or "log out" devices and require a user to re-authenticate unattended
8. Identify and implement the above measures systems with default passwords. Routers, switches, Web applications and administrative web interfaces, ICS systems, Telnet and SSH interfaces are among the systems that need examination
9. To reduce the brute-forcing risk of attacks implement lock-out policies. 9.
10. If there is no remote management requirement, Telnet and SSH should be disabled on the device
11. If remote access is necessary, configure VPN and SSH for the access device.
12. Configure telnet client-based certificate authentication for remote device management
13. Router-level filtering of egress and input. 13.
14. Report the internet service provider in case found any suspicious entries in routers
15. Maintain up-to-date computer antivirus.

16. Keep IoT devices, operating systems and applications up to date with patches and fixes.
17. Stopping and closing unnecessary ports and services.
18. All activities must be logged on the device.
19. To detect scanning attempts on critical devices/systems, activate perimeter device logs and monitor them.
20. DDoS prevention countermeasure:
21. Identify and prioritize essential services. Continuity Business Plan Development.
22. Use the appropriate system for detecting and mitigating DDoS attacks. Introduce appropriate intrusion and DoS prevention.
23. Ensure that the Intrusion / DDoS Prevention System includes signatures for detection of common DDoS tools launched attacks.
24. Keep a list of ISPs, network vendors, and security devices contacts and contact them as required.
25. Understand the current environment and establish the basis for daily network traffic volume, type, and performance.
26. Check traffic patterns and logs for traffic abnormalities, floods in the network (TCP, UDP, SYN, etc.), and application flooding. Application floods (HTTP GET)
27. Maintain and review webserver logs for malformed applications/traffic regularly.
28. If the ISP SLA includes mitigation services from the DDoS, which has to be informed to the employees so that the requirements to be forwarded to the ISP.

Conclusion: But countermeasures have been introduced and the only way of mitigating such attacks is utilizing precautions. The Mirai Botnet effects and misuses IoT devices for DDOS attacks or gives hackers the privilege of controlling such devices as they want.

3.7.3.4 H04 - Case Study 4.

Title: Necurs BotnetIntroduction:

Original Issue Date: September 24, 2019

Virus Type: Banking Trojan

Severity: High

The malware variants known as Necurs (Necurs Botnet, 2019) have been reported to spread. Themalware targets mainly windows operating systems and is famous for its functionality for spamand malware distribution. The malware mainly contains phishing URLs or malicious attachments via spear-phishing email and also via databases.

Objective: This botnet aims at the distribution through dating sites of Phishing Emails with phishing URLs and malicious attachments. It essentially attempted to deactivate the antivirus services and to remain on the windows. Since it is a botnet, the entire PC control is forwardedto the hackers. To pass such commands that might be used for the hacking purpose.

It has the following functionalities:

- Function to hide from detection by disabling components or other security features of the antivirus driver.
- Spread banking trojans, RATs, RAS, cryptocurrency miners, or info stealers.
- Stop the work for some time and then reset the infected hosts to apply new commands.
- Necurs botnet infected machines provide remote command and control server network connection to receive and operate commands accordingly.
- Take advantage of victim email IDs for spam-sending.
- Malware spreads that can start DDoS attacks.

Necurs has a rootkit kernel-mode capability which includes the driver of the kernel-mode anda user-mode component, giving the attacker maximum privileges. It has also modular architecture to spread various malware and, when necessary, perform different functions.

Sample Collection, compilation, and Implementation:

Network Connections:

It uses the DGA (algorithm for domain generation) to hide its activity and prevent detection. Every time a new domain is registered, the corresponding IP address of the C2 server is still obscured and the connection to the remote C2 server is then decoded by a bot. These DGA domains cannot be disconnected by this encryption.

For the generation of domains, the DGA algorithms are DGA double-borders which use 2 DGAs. The following are explained:

- DGA1 detects the environment of a sandbox and only generates 4 domains simultaneously.
- DGA2: generates 2048 TLD domains (top-level domains) covering 43 different TLDs and expires daily.
- It also has some hardcoded domains for calling a C2 server to serve as fallback domains.

IOC:

Malware Hashes:

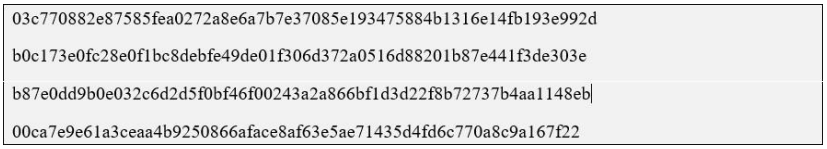


Figure 3.23: The list of complete files hashes, also available online.

Best Practices

1. Delete malware system changes such as creating files/registry entries/services etc.

2. Traffic monitor generated from the domains of the client machines and IP addressesreferred to in the Setup Section.

3. Avoid pirated software downloading.

4. Protect yourself against attacks by social engineering.

5. Antivirus Solution Scan infected system updated

6. Policy deactivation for autorun and autoplay.

7. Use limited computer privilege users or enable administrative access to administrator'ssystems with special administrative accounts.

8. Do not visit websites without confidence.

9. Do not download or open an attachment in emails received from unreliable sources ortrustworthy users inadvertently.

10. Implement a strong password policy and change passwords regularly.

11. Activate a personal workstation firewall.

12. Disabling unnecessary workstation and server agency services.

13. Change the default login credentials before production deployment.

Conclusion: It is another bot, tries to control the infected zombies and pass the commands. The planet spreads to bankers, ransomware, RATs, info stealers, or cryptocurrency miners allover the world. It uses the victim's email identifications to send spam emails and use them to attack DDOS. Few countermeasures were shared.

3.7.3.5 H05 - Case Study 5.

Title: Andromeda BotnetType: Backdoor

Severity: Medium

Introduction: The malware families called "Andromeda" (Andromeda Botnet, 2018) or "Gamarü"have been reported to spread. The malware is mainly designed to build a network of infected computers that was later part of Andromeda Botnet on the windows operating systems. The botnet is then used to spread other malware families associated with Andromeda. The malware is a modular bot, with plugin functions such as keyloggers, rootkits, team viewer, dispersion plugins, etc. Malware-generated infection

vectors include spelling phishing emails, drive-by- downloads, infectious cracks or keygens, removable drives, malicious links over social media(such as Facebook). The botnet was dismantled in close collaboration with public, private, andgovernmental organizations around the world on 29 November 2017 by the law enforcement agencies.

Objective and Capabilities: The aim is to scatter around the world and use Windows machines to target them. They probably spread ransomware and other spyware applications such as keyloggers, Trojans, and access to sensitive operating system information.

The malware can do the following functions:

- Capable of anti-virtual and anti-debugging technology

- Can be used as a start pad by distributer of other malware, such as ransomware (Petya, Troldesh, Cerber), trojan banking (Ursnif, Fareit&Carberp), malware DDos (Fareit, Kasidet), a spam bot (Cutwail&Lethic), backdoor, etc.
- The Avalanche botnet was used.
- It works as a backdoor that can get commands to download and run files from its controlserver, execute remote shells, or uninstall from the system.
- Its memory resident.
- Steals sensitive information such as information on the operating system, local IPaddress, serial number of the root volume.

• Allied: Wauchos, Gamarue, Backdoor.Andromeda

- **Sample Collection, compilation, and Implementation: Indicators of compromise:**

- **File system changes:**

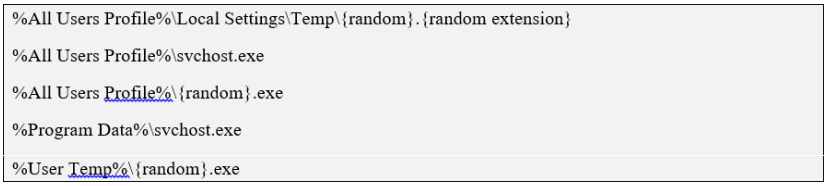


Figure 3.24: File system changes

- **Injects itself into the following processes:**

Malware creates a new instance of the below-mentioned processes to inject itself.

Figure 3.25: Inject itself into the process

- **Registry changes:**

Registry changes made by the malware are:

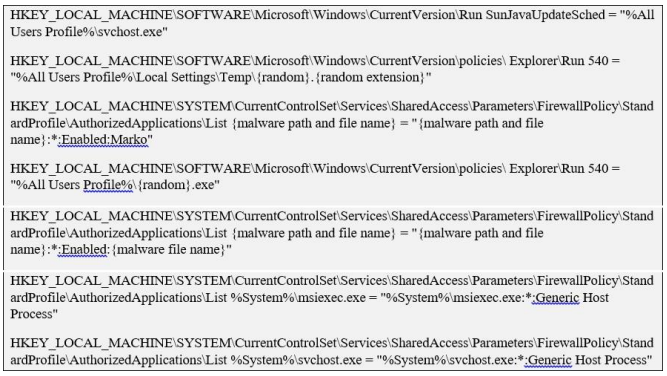


Figure 3.26: Registry changes

Network Connections:

The malware connects to the below-mentioned domains:

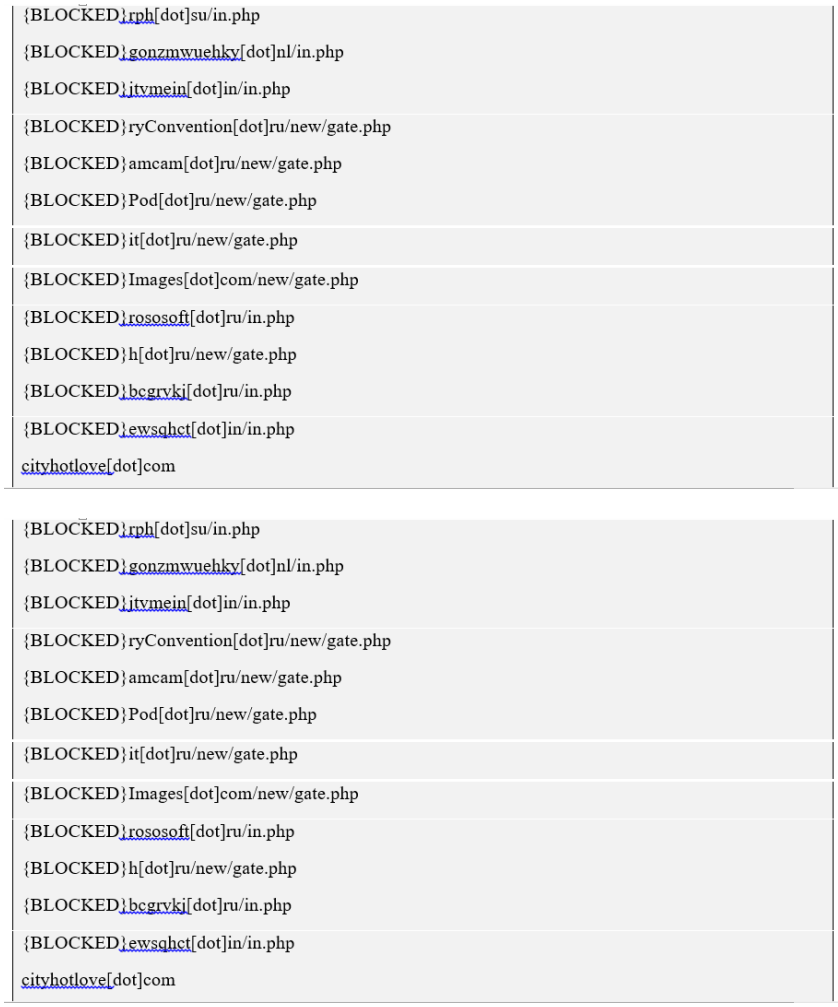


Figure 3.27: Malware Network Connections

3.7.4 Key Findings of the Hypothesis Experiments

When connecting via IPv6, it's not only the concept to change IPv4 to IPv6 in VPN Connection issues. The IPv6 is also characterized by many technical issues. In the firsthypothesis, the MTU values were mentioned, instead of the default value of 1500, between 1380 and 1450. It was clearly stated how to make changes, but most people donot know about these technical issues when they switch from IPv4 to IPv6. The further Hypothetical example states that, despite many security measures, vulnerabilities used in VPN products that are used worldwide create a direct backdoor.

The Mirai Botnet will impact and misuse IoT devices for DDOS attacks or give hackersthe privilege to control devices according to their preferences.

It was reported that the other botnet "Necurs Botnet" is spreading the malware versionscalled Necurs (Necurs Botnet, 2019). The malware targets mainly windows operating systems and is famous for its functionality for spam and malware distribution. The malware mainly contains phishing URLs or malicious attachments via spear-phishing email and also via databases.

The malware family variants named "Andromeda" (Andromeda Botnet, 2018) or "Gamarue" have been reported to be spreading. The malware is mainly designed to build a network of infected computers that was later part of Andromeda Botnet on the windows operating systems. The aim is to scatter around the world and use Windows machines to target them. They probably spread ransomware and other spyware applications such as keyloggers, Trojans, and access to sensitive operating system information. Based on the above quantitative and qualitative experiments, the MIX research approach was chosen for this research.

3.8 Proposed Model of Framework.

3.8.1 Objectives of the Study

The Internet is full of versatile malware and the demand for Secure VPN has increased considerably following the COVID-19 pandemic. The existing hardware and software are usedto implement the VPN, but the existing VPN services have been discovered that they are not sufficiently safe to deal with malicious attacks and it is even a long time to deal with an incident. In addition, the evidence for identifying the special point device for attacks in the Infrastructure is inappropriate. This proposed framework was designed to update and train the system to locate connections, attacks and provide the correct evidence to detect any attacks thatmight be applied.

3.8.2 Significance of Research

An Internet security service creating an encrypted connection between user devices and one ormore servers is a Virtual Private Network (VPN). VPNs can connect a user securely to the internet or public internet network of a company.

Companies normally use a VPN to provide remote employees with access to internal applications and data or to create a common network between various offices. In both cases, the ultimate objective is to avoid exposure to the open Internet of Internet web traffic — especially the traffic that contains proprietary data. When employees work on-site, they can directly connect to the company's internal network with their computer and mobile device. However, if an emp

loyee functions on a remote basis, his connection to that internal network must be made via the public Internet and his traffic may potentially be exposed to ongoing attacks and other sensitive data snooping. It keeps traffic from prying eyes safer to the crypt with a Business VPN or other security service.

3.8.3 Need of the Study

The traditional approach to IT to secure the company is using perimeter security – hence the popularity of private, virtual networks (VPNs). While VPNs in the past could effectively enableand secure remote access when the majority of work on the site was carried out in corporate walls, their use in a world involving the predominance of virtual workers has becomeincreasingly risky. During the reception of company data and applications necessary for their work, company VPNs gave remote workers network access, other network data also became vulnerable.

It is not always possible to 'trust' local network users, as can be seen in the many data violationsthat affect organizations all over the globe. By allowing remote users to run the network, a large attack surface remains open for hackers and attackers to exploit.

3.8.4 Benefit of the Study

Based on this research, there are various advantages for companies to implement the VPN. It alleviates all common VPN security issues. The following are the following:

Secure Networks

The hackers keep a track of the business activities via online applications and websites or popup's or hyperlinks to trace and analyze the activities performed in an organizational Infrastructure. Without much knowledge of IT the best solution is to implement a VPN in the network so that all the anonymous malfunctions might be blocked at the VPN server side and only the filtered data transferred to the Clients machines. It also creates an anonymity from thehackers and able to transfer data securely. This is one of the important aspect of this research.

Hide Private Information

Hackers can use a range of methods to intercept information that is sensitive for the clients. They can try to personify using such information, gain access to bank accounts, card details, and more. However, with a VPN, it can be benefit from high-level security like 256-bit encryption. All online communications, therefore, seem to everyone, who can find a way to intercept them, like nonsense, rude text, and characters.

Avoid Bandwidth Throttling

Bandwidth throttling is when the ISP or someone else intentionally slows down the internet speed to control the functioning of the network. Sometimes this happens when a user visit or engage in certain websites. When using a VPN, it can encrypt traffic from the connected devices. Others cannot be able to identify the websites that has been visited with encryption.

Since the throttling of the bandwidth is somewhat triggered by the websites or type of activityengaged in, a person cannot throttle the data from and into a device if an ISP cannot see them.However, during certain times of the day, a person can still throttle a data to free bandwidth for other users.

In many cases, employees and others who use third party internet connection are not throttledon their internet use but their data transmission can be disguised by using a VPN, and the possibility eliminated.

Get Access to Geo-Blocked Services

Another Internet Protocol (IP) address can be obtained with a VPN. IP addresses indicate the location of the device while browsing the internet, streaming content, or participating in otheronline activities. Some websites and services prevent users from certain countries fromaccessing certain or all of their offerings. Streaming services which cater to particular locationsare common.

It is also common to limit the way to use services while accessing Internet, for example receiving quotations and accessing more specific information on their services at certain business websites. If using a VPN, it is possible to restrict the Internet access for a particular location to the service tried to access.

If all information and services offered by the staff are to be fully accessed by websites, a VPNcan facilitate this as per the need.

Network Scalability

Although a private network can assist the company's, the network expansion cost can be prohibitive. When using a VPN, many employees and remote staff can be accessed at the sametime. A person can also run key applications in a cloud environment and access them via the VPN's secure tunnel.

This can include anything normally run on a desktop computer from email to full-sizeapplications. If employees connect to the VPN, they will be able to access a different computerto run the application they need. Every logged-in employee can access the VPN and the application. Adding additional employees means providing each new team member with morebandwidth and login credentials if necessary.

Reduce Support Costs

The opportunity to save substantial money in support services with a VPN setup that includescloud computing architecture is there. The performance and maintenance of the internal serveris typically the work of the internal IT staff with an on-site set-up, for example. This may require hours to check how well the server performs, if all employees get optimum results andif hackers or malware attacks.

Furthermore, when a problem is identified, it required more time to deal with the problem andthe consequences might occurred in an organization.

However, the service provider is responsible for all maintenance, performance controls, and security measures with a VPN. A large number of paying customers support their IT expenses,making their per-customer costs relatively low.

To ensure that this is the case, the services offered by the provider and the hardware they use are prudently monitored. More modern components and safety measures often lead to a bettercustomer experience.

3.8.5 The Theoretical Framework of the Present Study

The theoretical framework proposed must be configured based on hypothesis experiments andthe available IPv6 implementation of VPN. The current flaws or limitations based on problems that most of the organization raises need to be identified further. This proposed framework supports the elimination of existing vulnerabilities and problems and proposes a new framework to safeguard the whole Intranet model and network-wide data. The combined resultsare based on the different experiments carried out and carried out successfully.

3.8.6 Conceptual Model Framework

Our concept model framework is a design created to identify and eliminate the risk of the implementation of VPNs using IPv6. For detection, system training, and updating of additionalsecurity compliances, it utilizes advanced machine learning algorithms. In the proposed modelframework the list of activities included is as follows:

1. Identify and authenticate the host. In IPv6, add MAC (HIP - Host Identity Protocol, NDPNeighbor Discovery Protocol)
2. Secure transmission of data with anonymity (requests and responses).
3. Policy-based privilege escalation
4. Tracked and monitored. (Procedures for conduct) Creating packets to customize
5. Dataset Training

6. Updates and patch management

7. Analysis and testing.

8. Loop/Recycling.

3.8.7 Results/ Expected Outcomes.

1) Core implementation: This is the initial stage of the proposed framework in which a Linux platform core VPN implementation configured by IPv6 using Static or DHCPv6Server is implemented.
2) Compliance of security: Completed core implementation and settings. Furthermore identifying and implementing the Host-based connectivity and detecting the connecteddevices accordingly. In addition, the software has also a built-in network for customizing the packet which carries encrypted values to track devices and their path, even if connected to a proxy server or any routing protocol.
3) Patch management for security: This stage of patch management for security is very important since the organization needs personalized support and has various applications and protocol data types on the Application Layer, so it's important that datasecurity patches are integrated and that the network connections performance is testedaccurately.
4) Tracking and monitoring: the packet monitoring and tracking required for ensuring thecorrect transmission and confirming, based on integration of safety gaps and patches inthe core setup. Finding the log compliances necessary for response to the incident and further research purposes is also important.
5) Training Model and repository updation: This model refers to the algorithms for machine learning to train the databases/repositories related to trusted devices, malwareattacks, and botnet operations.
6) Testing and Analysis: The cost factors, performance factors, security factors, technical complication factors and other management factors form part of further assessment andanalysis of the proposed framework.
7) Results and Findings: The conclusions shall be based on the results of the proposed framework on basis of technical procedures and limitations and the overall benefits.

3.8.8 Proposed Model

The proposed framework model combines the implementation and analysis of the different phases involved in preparing a secured model.

Legitimate users as well as Botnet’s are constantly targeting an Organization's Network Infrastructure, however, organizations have been shown to apply the generic VPN process andto take security less priority due to the lack of technical, financial, and other internal organizations, thus compromising the infrastructure. In the proposed model, the goal is to provide an adequate means of implementing the VPN infrastructure, which is classified into six categories:

1. Authentication and Authorization

2. Application testing and integration

3. Privilege acceleration

4. Log Analysis, Data Collection, and Sanitization

5. Data Analysis and Synchronization of Trained Engine

6. Repository updates and Model Synchronization

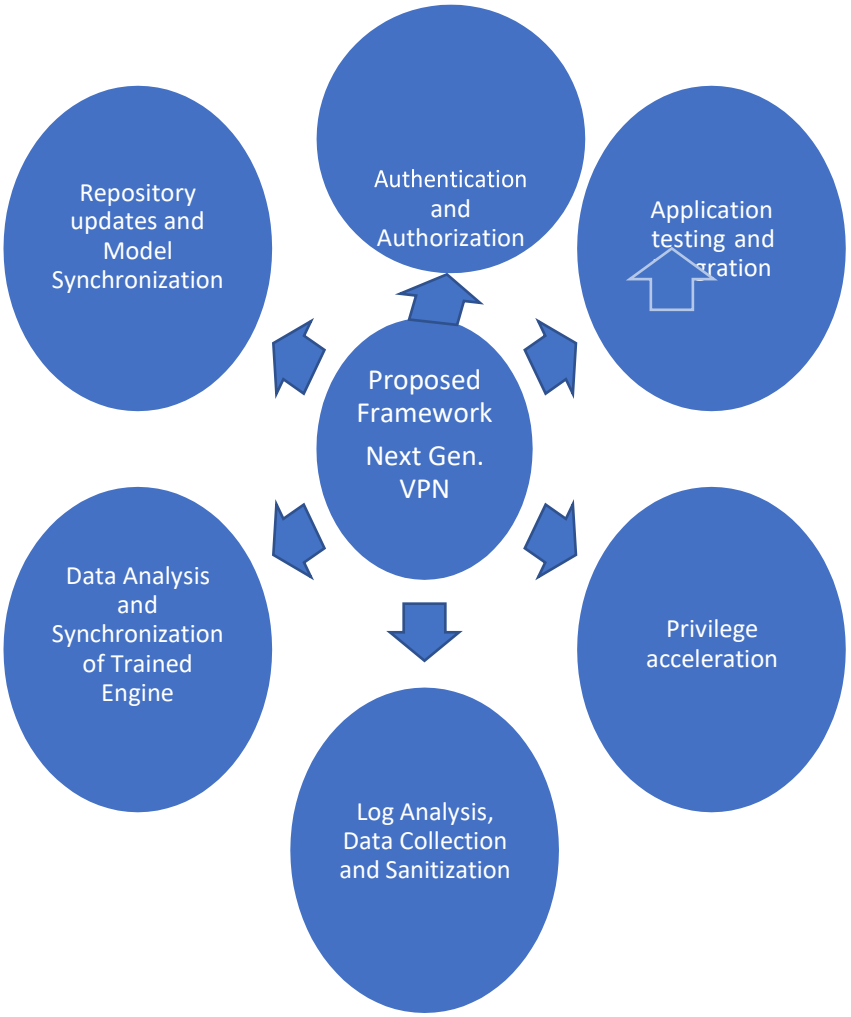


Figure 3.28: Proposed Framework of Next-generation VPN

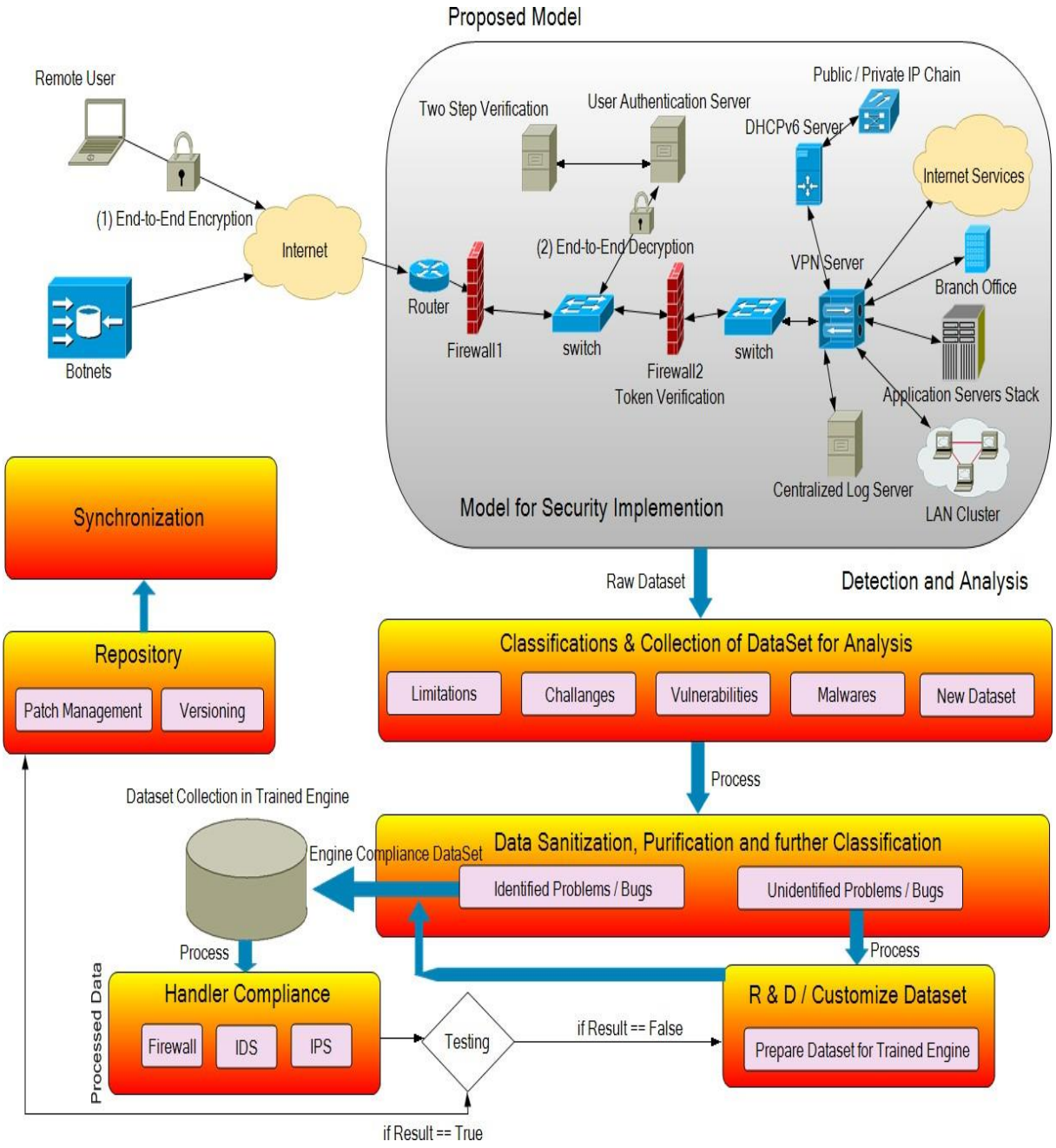


Figure 3.29: Proposed Model

3.8.9 Objectives of the Framework Implementation.

Step1: The first stage, as stated in the proposed model, is to ensure that the secure VPN infrastructure is implemented properly. While legitimate users and Botnets hit the VPN, generic VPN implementations lacked security policies, the first step proposed in the model wasend-to-end encryption between remote users and front-end routers.

Step2: Firewall-1 should prevent all irrelevant Internet requests, and for internal authentication, only VPN requests are to be processed and forwarded.

Step3: A user authentication server (UAS) (Authentication Server, 2019) is needed for applicationvalidation for internal authentication. The policy at this stage must determine whether the requirement comes from the trusted device or not, if the system is trusted, the request must beprocessed by UAS and the End to End asymmetric encryption applied is decrypted. If the device is not trusted, then the two-stage verification model will be implemented and the user and manager will be notifiable. **Step4:** If everything goes smoothly to complete end-to-end decryption, a TLS token for specific support needs to be created for UAS servers. The additional token was forwarded for the token verification to Firewall-2. The concept is based on the customized package and techniques of token passing.

Step5: Now if the token is checked then only the request must be transported to the VPN serverand sent via static or DHCPv6 servers to allocating IPv6.

Step6: Now the public/private IPv6 chain IPv6 address with the combination of MAC addresson the confident device is allowed according to the VPN server requirements.

Step7: Once IPv6 has been allocated to the end-user, they can access services such as Application Servers Stack, Branch Office, LAN cluster, and other web-based services that arebased on the specific user policy. Once the user is approved, in legacy VPN, there is a privilegeto access all network services and ports. However, it has been mitigated in this proposed framework.

Step8: All activities are registered on the centralized log server, and behavior analysis and detection techniques are further processed.

Step9: The framework should be trained with known samples and samples collected.

Step 10: The framework is tested using live wild samples.

Step 11: A limited and future scope analysis of the proposed framework.

3.9 Algorithms used for Building the Framework

The entire Framework is based on the combination of various stages to accomplish the expectedoutcomes.

3.9.1 The Final Algorithm

The final Algorithm is completely based on the six major stages of the proposed framework.Let us assume the NGVF is the multifactor set

NGVF={f(a) U f(b) U f(c) U f(d) U f(e) U f(f)}

Where:

NGVF=Next Gen VPN Framework.

f(a) →Authentication and Authorization Analysis

f(b) →Application Testing and f(c) →Privilege Acceleration Analysis

f(d) →Log Analysis, Data Collection, and Sanitization

f(e) →Data Analysis and Synchronization of l Synchronization

Figure 3.31: Elements of NGVF (Next Generation VPN Firewall.)

```
f(a)

{

Module
authorization(visitor)

{

If user(credentials) ==
exist



{ td =              // where td = trusted Device

true}

Else

{td
=False}



If td == true

{user = defined

permissions, groups and

policies}Else

{user = guest policies, Notification and admin alert, process to
identity management}

}
```

Figure 3.32: Authorization Module

Authentication Module:

This module includes the user activities to interact with identity management which identifies the trusted/ legitimate systems and generates the token. It also helps to generate the authenticated permissions to grant access. All the activities are logged and updated in the database.

```
Module Authentcation()
{
        Userlogin(token)
        {
                If token == validated
                {
                        User = encrypted_tunnel(https, IPv6, end-to-end encryption,
                        permission to
application access)
                        Log = user activites()
                }
                Else
                { User = encrypted_request(2FA, end-to-end
                 encryption)Log = user activites()
                }
}
```

Figure 3.33: Authentication Module

3.9.2 Application Testing and Integration Analysis

This section includes the second phase of the framework, where user anonymity, applicationpermissions, and integration are required.

```
F(b)
{
Module ATI()
{
User_anonimity(IPv6, https)
{
Authenticated_user =
allotment(anonymous_IPv6_with_LAN_restric
tions,Policy_based_specified_services)
}
Application_permission_integration()
{
Grant_access = true
Integration_of_resou
rces = true
}
Application_access_logs(user_activity, Application_activity)
{
If (grant_access == true) AND (Integration_of_resources ==
true)Log_user = user_activity()
Log_app = app_activity()
```

Figure 3.34: Application testing and Integration Analysis

3.9.3 Privilege Acceleration Analysis

This module is the heart of the framework. It takes cares of the privilege acceleration of the entire system. All the available listed applications, integrated applications, trusted networks and their resources, integrated trusted devices, policies, and permissions are mentioned in thismodule.

```
Return integrated_app_list()
}
Authenticated_networks()
{
        Return trusted_network_details()
        }
Policy_allotment()
        {
        Return policies_permitted()
        {


Policy_firewall_idps()
        {
```

```
Return integrated_app_list()
}
Authenticated_networks()
{
        Return trusted_network_details()
        }
Policy_allotment()
        {
        Return policies_permitted()
        {


Policy_firewall_idps()
        {
```

Figure 3.35: Privilege Acceleration Analysis

3.9.4 Log Analysis, Data Collection, and Sanitization

In this phase, it is the process of behavior analysis of the entire framework. It includes the classification and collection of the dataset, data sanitization, purification, and further classification of the activities performed in the framework. Further classified data

```
f(d)

{

Module Loganalysis()

{

datacollection(logs)

{

Centralized_Log_server(application_logs,firewall_logs,network_logs,User_activities,device_details)
```

```
{
Sanitized_data = Centralized_Log_server(logs)
}
Data_purification(logs)
{
Filtered_data = Sanitized_data() + gathered_data()
Return filtered_data;
}
Data_classification(filtered_data)
{
if (filtered_data == identified_issues)
{
Activity_logged()
}
Else
{
Trained_dataset()
Activity_logged()
}
}
Trained_dataset(filtered_data)
```

Figure 3.36: Log Analysis, Data Collection, and Sanitization

Data Analysis and Synchronization of Trained Engine Dataset

This phase includes the behavior analysis of filtered data/ logs, updation of the dataset, andsynchronization with the trained engine of all the activities.

f(e)

```
{
Module data_analysis
{
Behavior_analysis(filtered_data)
{
if (filtered_data == identified_dataset)
{
Activity_logged()
}
Else
{
Prepare_dataset()Trained_dataset()Activity_logged()
}
}
Update_dataset(filtered_dataset)
{
New_dataset = Trained_dataset(filtered_data)
}
Update_firewall_dataset(filtered_dataset)
```

{

Firewall_dataset = trained_dataset(filtered_data)

}

Firewall_syncronization(update_firewall_dataset())

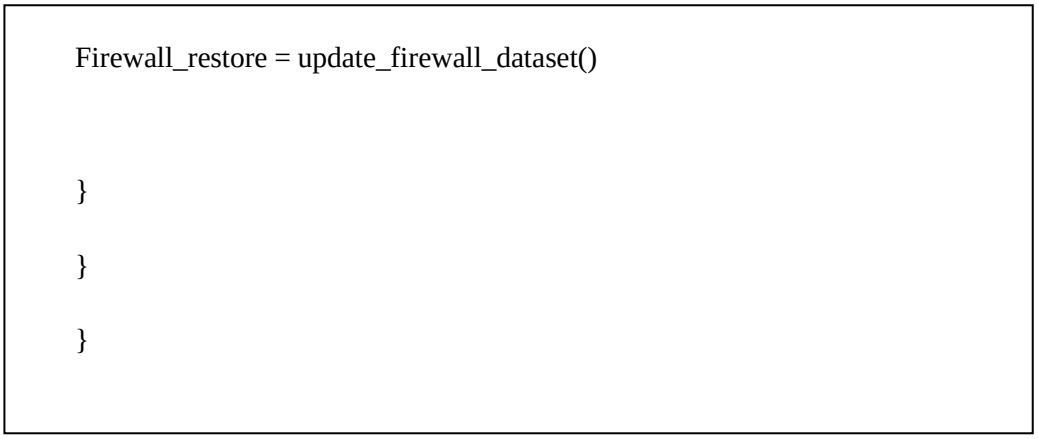


Figure 3.37: Data Analysis and Synchronization of Trained Engine Dataset

3.9.5 Repository updates and Model Synchronization

This is the last phase of the framework, it supports updating the repository which includes patchmanagement or versioning, etc. The entire model synchronization takes place and finally getsthe trained dataset for future evaluation.

```
f(f)
{
Module Repository_update()
{
Patch_management(new_dataset)
                                        {
Update_patches(new_dataset)
}
version_management(new_dataset)
{
        If (dataset ==exist)
{
Update_new_version(dataset)
                }
        Else
                {
                        Create_new_version(dataset)
}
}
```

Figure 3.38: Repository updates and Model Synchronization

CHAPTER 4: ANALYSIS AND INTERPRETATION

This chapter explains how the various experiments carried out for the proposed frameworkare analyzed and interpreted.

4.0 Building, Analysis, and Interpretation of the Framework.

There are many experiments, analyses, and interpretations required to build the proposed framework. A range of tools for implementing and analyzing the impact of the implementationsection, utilities, network resources, and other relevant IT infrastructure.

4.1 Identity Management – Concatenate Media Access Control (MAC) address toIPv6 and testing with various protocols

This section examines the principles governing the operation of the IPAC system and discusses several aspects behind the system design. Here it has been concentrated on theoryand architecture; details about the actual system implementation.

This experiment, it demonstrates the ability to link network administrators and incident respondents at a specific point in time, using only the existing networking mechanisms inplace on many corporate networks, with an IPV6 address to a specific media access control (MAC) and physical network interrupts. This is achieved by three steps: identification of all active network IPv6 addresses, correlation of each IPv6 address with a specific Layer 2MAC address, and correlation of the MAC address with a certain Ethernet Switch Port.

Step 1: **Identify IPv6 Addresses**

The discovery of IPv6 addresses in a local network is made simple by an existing mechanism in all IPv6-activated networks: the NDP (Neighbor Discovery Protocol, 2020). If a node wants to send the traffic to a given IPv6 address, it first needs to find the MAC addressof the target (or of the appropriate gateway router). This is achieved by sending an NDP Request to the target requesting its MAC address, in which the NDP Advertisement with the appropriate identification will respond. The source node does not know the target layer2 information and these requests are sent to the appropriate

Solicited Nodes multicast group, which means that the NDP solicitation packet is sent to each of the nodes on the network as shown in Figure 4.1.

NDP's behavior, which allows us to identify every IPv6 address in a local network. It must beshown in an NDP Request before an address can be sent to the address. 4 By listening to all NDP traffic and parsing source and target addresses, a complete list of each IPv6 address that is used on the network can be compiled. Because the NDP requests are multiplied for everyone,this can be done entirely passively. The process cannot identify the existing, but unused secondary IPv6 addresses on a host interface. These cases are irrelevant since the addresses would never be seen by an investigator in log files or packet captures.

Step 2: **IPv6 -> MAC**

The next step in the system is to correlate each IPv6 address with a specific MAC address once IPv6 addresses are identified. Other IPv6-enabled nodes via NDP are already performing this procedure. By just sending our NDP requests to the IPv6 addresses observed, It can generate an NDP advertisement that contains the appropriate MAC address value as shown in Figure from the target. This also enables us to validate that the IP address is present on the networkbecause there will be no response from a missing address.

To determine the entire Layer 2 status of a local network at a time, the continual monitoring of the existing multi checked NDP traffic and acquisition by NDP Solicitations of MAC addresses. The event will be marked with NDP traffic and monitored through our system whennew nodes are connected to the network or IPv6 addresses. The list of currently active IPv6 addresses on the network will be kept up to date by regularly resending NDP requests and the target may be considered to no longer form part of the NDP advertisement if valid NDP ads are not returned. Historical records of the status of the network can be kept simply by capturingthe times when the address is attached and left.

Step 3: MAC -> Port

MAC addresses aim to give the individual Network Interface Cards (NIC) a unique global identification; however, there are several widely known methods in network traffic for the change or spoofing of such identification (“MAC address”,

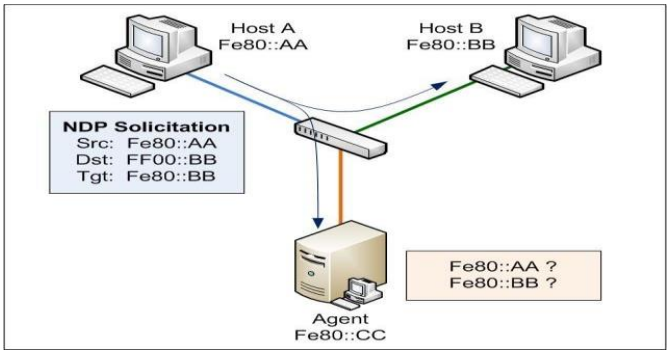


Figure 4.1: Identifying IPv6 Addresses in Multicasted NDP Solicitations

2020). Because MAC addresses are not immutable, it is not simply possible to rely on a MAC address to identify a system or user to comply with forensic requirements. Instead, the MAC address must be associated with some method to a particular physical device or place. This is achieved through the use of managed Ethernet switches and the Simple Network Management Protocol (SNMP).

Ethernet switches maintain an internal table to match the MAC addresses in physical ports based on the traffic they observe so that they can provide layer 2 switchings. This facility often called the Content Addressable Memory (CAM) or Bridge forwarding table (Bridging, 2020), provides an almost real-time view of the connection linked by the switch to Mac-Port pairings.Many popular managed access layer switches for enterprises provide a mechanism for monitoring a remote manager via SNMP status of (and subsequent changes to) the CAM table. When a MAC address connects to the net, is moving to a different port, or becomes dormant in the network, the switch notifies SNMP Traps or Inform messages to the event manager andassociated physical ports as shown in Figure 4.3.

Based on this feature, the physical switch ports used by any MAC address can be recorded over a while. When the MAC addresses enter and leave network, the manager is notified of the check-in or check-out by the switch. These notifications can be used by the manager to create a picture of the network state of layer 1-2 at any given time. As long the MAC is traceable throughout the network while all Ethernet connections within the network of the organization. These SNMP messages are produced.

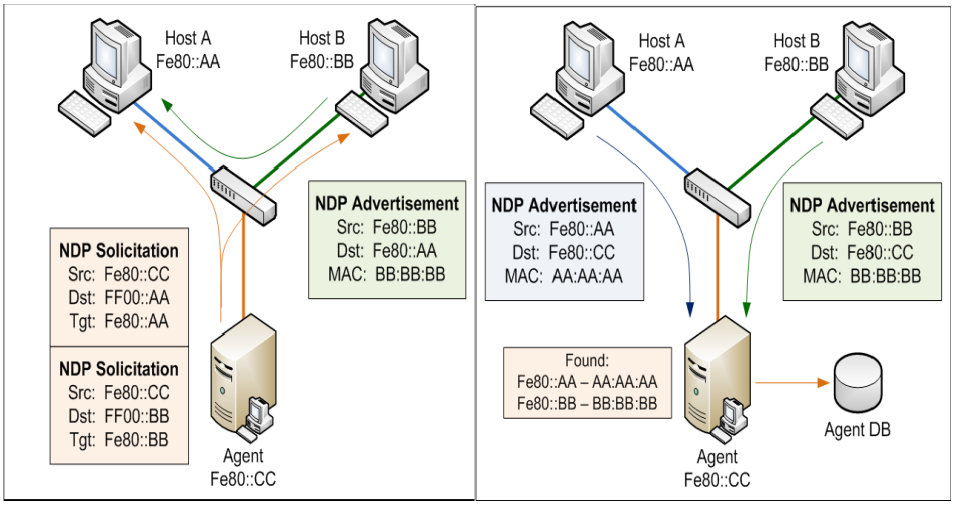


Figure 4.2: Sending NDP Solicitations to verify IPv6 address and request MAC addresses (left). Hosts respond with NDP Advertisement containing MAC addresses (right).

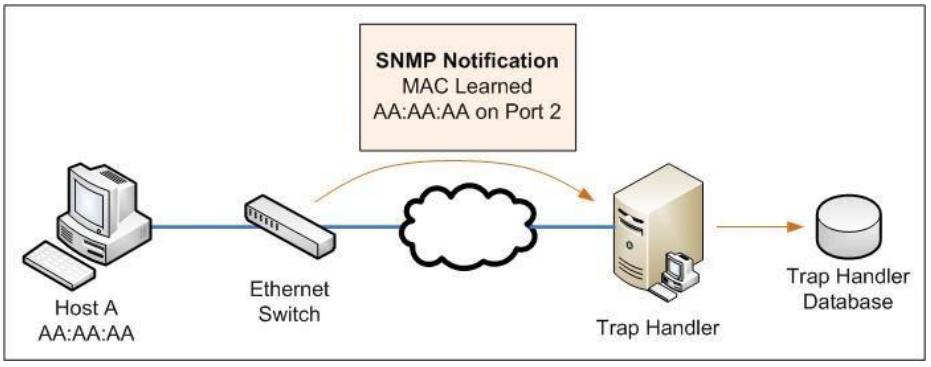


Figure 4.3: Receiving SNMP Bridge Forwarding Table Notifications from Ethernet Switches When Nodes Connect (top) or Disconnect/Timeout (bottom).

4.1.1 Modules

The system shown offers the functionality described by four separate modules in the previouslydebated procedure. Figure 4.4 illustrates the communication between these modules, while Section 7 details the actual application of each of those modules.

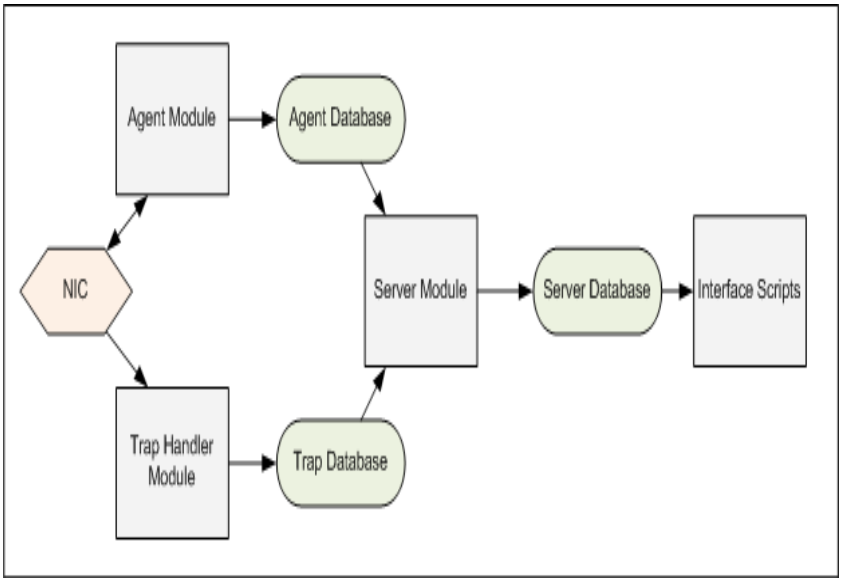


Figure 4.4: Flow of Information between IPAC Modules (grey rectangles) and Databases (green ovals).

Agent Module

The first module, the IPAC Agent, is responsible for the system's NDP functions. It monitors the network's multi-cast NDP traffic and scans packets for the IPv6 addresses for both source and target. Once addresses have been observed, the Agent generates its own NDP Request packets to validate and receive the corresponding MAC addresses. When an IP address is observed and validated, the server module saves a sighting record with the appropriate MAC address and timestamp. The agent also keeps a list of IP addresses it currently knows to be active in the network. This list will periodically check and re-validate addresses that have not been seen recently in other NDP traffic. If the address cannot be re-validated (after several attempts there is no valid NDP Advertisement received), the address is considered to have leftthe network and logged for future Server Module processing. Due to the operation of NDP traffic in Layer 2, a

separate agent is required for each domain on a network of the organization,although a few changes in Section 9.2 may reduce that restriction.

Trap Handler Module

The second module handles all system functions relating to SNMP Trap. It reads messages from Ethernet switches to the organization's access layer, which signals that Mac addresses from the CAM table of the switch are added or removed. Once these notifications have been received, the server module will store them for subsequent processing. Every switch within thenetwork of the organization must be set up for CAM table change notifications. As SNMP traffic can be routed, only one Trap Handler can serve the whole network of the organization. But if server or traffic charging could become an issue for a company, multiple Trap Handlerscan be used to serve various sets of Ethernet switches.

Server Module

Most of the functions of the IPAC Server module are to convert raw sighting and trap recordsinto usable and queriable data. This set of scripts generates a timeline of all IPv6-MAC addresses and MAC-Port pairings, which summarizes every single sitting in a single recordeddatabase with the addresses and ports involved and marks the start and end times of the session.

The module starts by querying unprocessed sighting records in agent databases as shown in Figure 4.5. After it has been obtained, this is condensed into synopsis sessions marking the startand finish times when the network saw an IP-MAC address pair. The sessions are added to theserver database and updated to reflect the new data by any existing sessions. The server performs the same function with any unprocessed SNMP trap records once all agent records are processed. Those records are linked to each session's starting and ending times and the resulting data set is merged into the existing server database. Sighting and trapping records willbe deleted from each database after processing by the Server module, and discarded so that olddata can never be recycled.

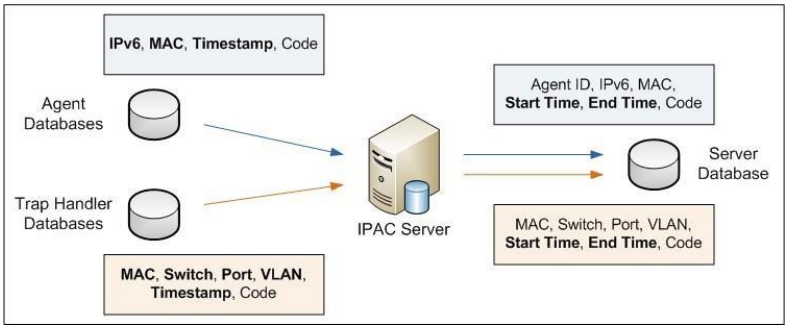


Figure 4.5: The IPAC Server Obtains Records from the Agent and Trap Handler Databases, Correlates andSummarizes These Records, and Stores the Summary Records in the Server Database

Interface Module

The final module includes a set of scripts that provide multiple methods for accessing and interacting with information maintained by the server module to network administrators and incident reporters.

The first script generates an IP address graph used over time within the network. This interactive view shows when an IP address is active, its associated MAC addresses, and the exact time it was first and last seen. This tool can also highlight instances of conflicts betweenIP-MACs (more than one MAC claims the same IP address).

There is another script for manually searching the server for specific data. A terminal-based utility. This script enables the user to retrieve a broad range of Server information, including IP Macs, IPs used by a Mac within specified time limits, the physical port used for a given IP or MAC address, current IP or MAC usage status, and instances of MAC conflicts.

A text-based log file with IPv6 addresses will be processed by the third script. The script will check the server for every IPv6 address in the file and will try to determine the associated MACand Ethernet switch addresses at any point in time. A small summary of the results is appendedto each line in the file containing the IPv6 address.

The final script provides the same functionality as the previous one but is applied instead of the text logs to binary packages. As the packet capture file cannot be modified without the original data being damaged, the results are given in a separate text file.

4.1.2 Other Considerations

Sending Unicast vs. Multicast Solicitations

One main point in the design of the agent system was whether NDP requests from the agent module should be sent through unicast or multicast packets. Unicast Requests havethe benefitof reducing bandwidth and the consumption of host resources. If the Destination MAC addressexists in the CAM table of the Ethernet switch, a single host will forward the frame of that packet to that host. On the other hand, the multicast requests provide greater visibility at the expense of bandwidth and host resources in the network state. Every multi-package is sent to and received by all hosts on all Ethernet links (though if the host correctly implements its NICdrivers, the packet will not be processed beyond the Destination MAC address). Since all hostsreceive the multicast packet, it is possible to detect scenarios with the same IP v6 address fromtwo or more hosts (each will send its own NDP Advertisement). Unicast requests from NDP are missed because in most cases they are only received by one host. With these considerationsin mind, it’s decided to implement the behavior of the system in a way that balances network visibility needs and minimizes resource consumption.

Whenever the agent is required to perform address finding, multicast requests are generated: all addresses of MAC claiming to be "owned" are identified. To ensure no other MAC addresses claim to also have that IP address, even though a MAC address is known to use the IP address in question. Multicast solicitation is required.

Scenarios for this are cases where an Agent observes an IP address that he or she already doesnot know, as well as when he or she observes an IP/MAC association that differs from what he or she already knows. In addition, a multicast Request for each NDP request observed on the network is forwarded to the Target and/or Destination IP address. The agent should therefore have the same view of the network as the host. Whenever an agent needs to perform a verification of addresses, Unicast applications are generated. In this case, the agent already knows the IP and MAC addresses. The Agent will verify that the IP is used by the MAC with a unicast Require, which [still] activates on the network. This is done in scenarios that includechecking that the host remains on the network after a time off and verifying an IP-MAC addresspair that is regarded as an NDP Request Source.

Handling of Conflicts and Spoofing

1. Conflicts

To fulfill the forensic requirements of the IPAC system, it must be sufficiently resilient to dealwith situations in which conflicting indicators exist. A conflict arises in this system when twoentities claim that they have the same identifier or that they have control of it. There are two possible conflicts within the IPAC system: two different MAC addresses claim the same IPv6address, or two physical switch ports claim the same MAC address. The same MAC address isthe same IPv6 address. Although these conflicts infringe IP and MAC standards, the productionnetworks are still liable for defects, software defects, and a spoofing attack (discussed next).

The IPAC system is not able to prevent such conflicts. Its function is instead to recognize andrecord the instances in which they occur. Every single session is recorded when two MAC addresses claim the same IPv6 address or when two ports claim the same MAC address. Afteran investigator has had time to query information on the IP-address system, any conflicts that exist at a time specified are detected by the scripts provided with the Interface module and theIDs for the two parties to the conflict have been provided. In such cases, only one MAC addressor port may not be defined as the owner positively by the system, but the range ofpossibilitiesmay be decreased for the two. It is up to the researcher to determine exactly which methods they want.

The behavior of an address detection time complicates the detection of an IP address conflict through an interface module script (if an IPv6 address leaves the network immediately after it has been verified by the IPAC system, its departure will not be detected until 30 seconds later).The script requires the IPAC databases for all addresses matching the given IP in the window within 30 seconds before and 30 seconds after the specified time to ensure that no hosts are overlooked. Each person is reported as the possible owner if two or more are found. If only one

was known at the exact time (the others started or ended in the ±30 second window), the priorityresult will be stated.

This is the desired behavior but in certain scenarios, it can cause false positives. If less than 30seconds after/before the interrogated time a conflict started/ended, the two conflicts will be reported although the conflict doesn't exist at the exact time required.

Even when no actual conflict has taken place on the network, false positives may occur. Examples could include therapid unplug and plugging of a host into another physical switch port (MAC-Port conflict) or the movement of an IP address to another device as part of a high availability mechanism (IP-MAC conflict). In these circumstances, it would result in a conflict report to query the IPAC database within 30 seconds of a timestamp. This behavior is preferable to that that could lead to false negatives, despite this situation. If a network conflict existed, a false negative would happen, but only one (the wrong) node was viewed and reported because of the IPAC system's address detection time, which results in an incorrect host being assigned a log entry or a captured packet. A false positive does not involve a particular host but draws the attention of the investigators to a small group for more research.

2. Spoofing

The spoofing of IP and MAC addresses by a malicious party is one reason for conflicts in IPACsystems. IP address spoofing occurs where a host with a MAC address claims ownership by another host of an IPv6 address. The IPAC system uses a mechanism to reduce certain spoofingattacks. The IPAC Agent will send a multicast NDP solicitation if it observes the known IP address with a different MAC address and waits until the NDP advertisements come back before accepting the new pair. This prevents the system's data from being spoiled by an attackerwho injects only spoofed NDP traffic into the network; the new IP-MAC address pair will be simply ignored if the attacker does not answer the NDP request. If the spoofing system does, however, answer the request, the addresses are registered and conflicts are treated as describedabove. In such cases, it is still possible to track the place of the spoofing host through the physical switch port records of the system.

Spoofing MAC address occurs when a host uses a different MAC address from the hardware manufacturer. Although a switched network can be wrong, one host can use a MAC address on the same network already in use by another host. In these scenarios, the Ethernet switch observes the address on a new port and generates a MAC Learned IPAC system notification that indicates the physics of the spoofing host. Any traffic after the original host causes the

commuter to generate another notification that the original port has learned the MAC. The records are recorded through the IPAC system to handle conflicts, as described above. For an attacker, certain network topologies may spoof SNMP notifications into the IPAC system, which leads to corrupted data. The separation of management traffic in a separate network andthe strong authentication of the SNMP messages, as discussed in Section 5.3.4, can prevent this.

3. Time Synchronization

All devices that generate IPAC data must have synchronized clocks for the IPAC system to produce exact results. This applies to all Agents, Ethernet switches, SNMP trap generation network devices, SNMP Trap Handlers, and every device generating a log file or packet capture. Where two devices have non-synchronized clocks, MAC-IP records, MAC port records, and log entries may be misaligned to reduce the system's precision. Scenarios with anover 30-second misalignment could lead to a failure to identify a host, or worse, to misalignment to another host.

The Network Time Protocol (NTP) (“Network Time Protocol”, 2020) offers a standardized method for synchronizing time to network devices. Implemented within the system (not withinIPAC modules) host or device, this protocol uses a centralized time server to keep the internalclocks in hundreds of microseconds on all devices within the organization.

It should be noted that simplicity over security was preferred in the demonstration network when implementing NTP to minimize system complexity. The only safety consideration for the basic NTP configuration use client-initiated updates instead of server-initiated broadcasts (preventing rogue NTP servers from affecting the times on our systems). NTP implementationsin the production networks must implement strong update authentication via symmetric or public key encryption in addition to this fundamental safeguard (“Network Time Protocol”, 2020).

4. Security Issues in SNMP

SNMP version 1 (SNMPv1) traps are used to relay Ethernet switches' MAC Address Tablemodification events. Several security issues have already been widely discussed

with this version of the protocol (SNMP, 2020). However, SNMPv1 has been selected for this demonstration system, because it is easy and compatible with the available hardware and software for the laboratory. However, the latest protocol version, SNMPv3, which offers

much stronger authority and manager's authentication and encryption of messaging, is strongly recommended in a production network. That means all Ethernet switches of the organization have SNMPv3 functions which might be of concern to some organizations and those with older hardware. SNMPv3 encryption functions can only be used on those with export-controlled crypto-software images for some Cisco Catalyst models. If Ethernetswitching hardware is compatible, it would only require modification of the codes of the Trap handler module to implement SNMPv3 with this system.

4.1.3 IPv6 Operations Survey

A survey was conducted to find exactly how the IPv6 Protocol works between systems of the various vendors on the production network before the construction of the IPAC systemwas completed. This survey examined three main aspects of IPv6 behavior on popular OSsystems that directly affect the operational requirements of the IPAC system: the behavior,the use of multiple IPv6 addresses, and the use of IPv6 Privacy Extensions. Outline the methodology used to test each OS examined during the implementation.

Testing Methodology

1. Neighbor Discovery Protocol Behavior

The first part of this survey aimed at examining the way NDP traffic was generated and incoming NDP packages managed by every operating system. This behavior was determined through the creation of NDP Requests and Advertise for the target host, monitoring the resultant IPv6 traffic on the network, and monitoring changes to the NDP cache table of the targeted host.

The test network in this section included an Ethernet hub, a test workstation that produced NDP packets and monitored all network traffic, and a host running the

operating system. The targetedhost existed either as a physical workstation system or as a VirtualBox.

The host network interface card was connected directly to the wired network when a virtual host was used. Furthermore, the physical host running the VM did not have IPv6 functionalityenabled and left only IPv6 traffic nodes for the target system and test workstation. In each NDPtest scenario, an additional router is attached for long enough to produce Router Advertisementpackets that contain the global and unique local IPv6 addresses of the testing network (this generates the target host addresses in those scales). This network topology is illustrated in Figure

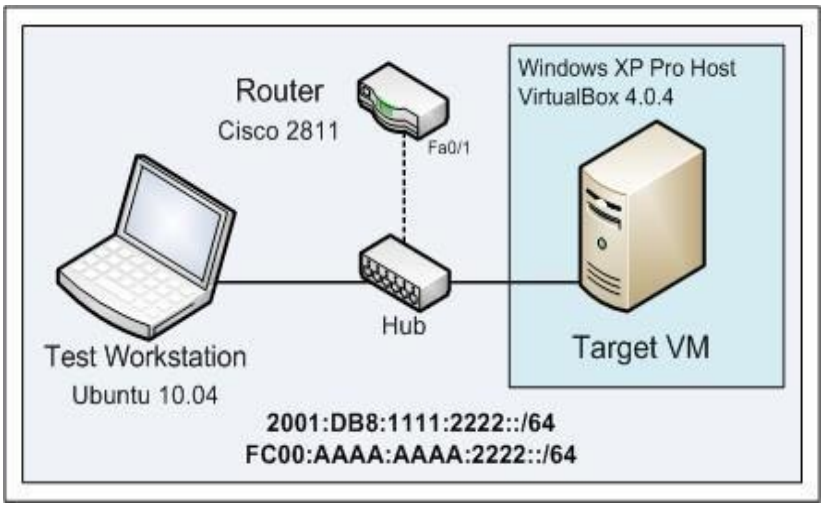


Figure 4.6: Network Topology for NDP Behavior Testing.

The purpose of this test was not to examine exhaustively how every host reacted to all the NDPscenarios. Instead, it focused on the normal protocol operation and simple cases of incoming packets slightly differing from the standard. The tool for generating traffic for testing the NDPbehavior of the targeted OS was built with the standard library and dpkt module of Python. Asthis tool was not intended as a complete protocol fuzzer to test every option in the generated NDP solicitor and advertisement packets, it tested only packet field combinations of select andall possible scenarios in which a single packet field differs from an IETF RFC4861 correctly constructed packet.

The testing results of the following scenario about the target host operating system are as follows:

Generation of NDP Solicitations

1. The options and flags of IPv6 and ICMPv6

2. Identify the Source IPv6 address

(Target's the same scope. Address of link-local scope.)

3. In multicast applications, list the values used for IPv6 and Ethernet destinations (Multicast/communication all-nodes. Multicast node requested.)

4. Identify the ICMPv6 datagram that has the Source Link-Layer Address option.

Generation of NDP Advertisements

1. The list of options and flags of IPv6 and ICMPv6.

2. The list of IPv6 addresses of source machine. (Solicitation target address. Address of link-local scope.)

3. The ICMPv6 datagram used the Link Target Layer Address option.

Handling of Incoming NDP Solicitations

1. The system adding a source address to the NDP table based on the Request.

2. This Request Package reciprocal with the NDP Request for the test system.

a. The malformed requests are the system will respond.

b. Field ICMPv6, not 0.

c. ICMPv6 incorrect checksum

d. Missing option ICMPv6 Source Link-Layer Address

e. Incorrect option for ICMPv6 Link-Layer Address

f. IPv6 All Nodes Destination (not Solicited Node)

g. Incorrectly requested IPv6 Node address destination

h. Field limit of IPv6 Hop is not 255

i. Flow Labeling field IPv6 not 0

j. Field IPv6 Traffic class field, not 0

k. Ethernet Destination to Broadcast

l. Incorrect Solicited Node Destination

Handling of Incoming NDP Advertisements

1. The system accepts unsolicited NDP Advertisements when no cache entry exists

2. The system accepting (requested, non-free) malformed is advertising

a) ICMPv6 Code field not 0

b) ICMPv6 Checksum incorrect

c) ICMPv6 Source Link-Layer Address option missing

d) ICMPv6 Source Link-Layer Address option incorrect

e) ICMPv6 Flag field combinations (Router, Solicited, Override)

1. Router only

2. Override only

3. Not Solicited (none)

4. Override and Solicited

f) IPv6 Multicast to All Nodes

g)IPv6 Multicast to Solicited Node Address

h)IPv6 Source not matching ICMPv6 Target

i) IPv6 Hop Limit field not 255

j) IPv6 Flow Label field not 0

k)IPv6 Traffic Class field not 0

2. Use of Multiple IPv6 Addresses

The next part of the survey looked at the use of IPv6 addresses by each operating system whenthe host is on a multiple IPv6 prefix and scale network segment. This has been achieved throughICMPv6 ping traffic generated by the targeted system and IPv6 addresses of the Source used for network traffic.

As before, the network used by this scenario included the targeted host (both physically and virtually) and the testing plant, both connected by an Ethernet hub, which sniffed network traffic. The network also included a second IPv6 segment that included virtualized CentOS and Windows Server was connected by IPv6. As in previous years, all virtual machines had no IPv6functionality on physical hosts.

Each Network segment had three IPv6 scopes with a prefix of individual ICMPv6 router advertisements (public, single-local, and link-local). Figure 4.7 displays the topology of this network.

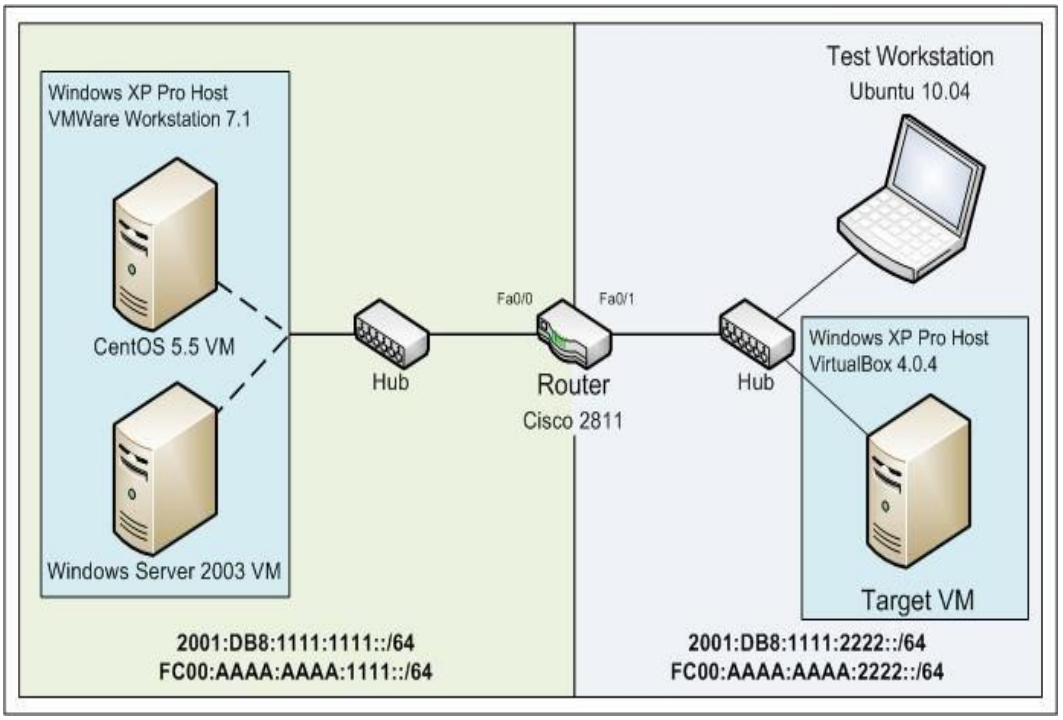


Figure 4.7: Network Topology for Multiple IPv6 Address Use and Privacy Extensions

Implementation Testing.

The tests attempted to respond to the following questions in this scenario:

1. The system will generate a prefix address plus a link-local scope for each router advertisingprefix.

2. The source addresses are used when pinging nodes in the same network segment.

a) The address prefix of the source matches the address prefix of the destination.

3. The source addresses used when pinging nodes in another network segment

a) For router requests from the NDP.

b)Ping to destination traffic for ICMPv6

3. IPv6 Privacy Extensions Implementation

The final part of the survey looks at how the host performs stateless address auto configuration privacy extensions in IPv6. This has been determined by checking and adjusting the confidentiality extension configuration of the target system, generating traffic from ICMPv6 Ping, and observing source IPv6 addresses that have been used when packets cross the network.

This section used an identical network with a virtualized or a physical target host, test workstation, Ethernet hub, IPv6 router, and a separate segment that includes VCs. The networkwas used in the Multiple IPv6 Addresses section of the survey.

Figure 4.7 illustrates the topology of this network. The testing attempted to reply to the following questions in this scenario:

1. The IPv6 privacy extensions be supported.

2. The IPv6 confidentiality extensions enabled by default

3. The IPv6 privacy extension will be used with enabled/deactivated.

4. The parameters which are default.

5. The process to change the parameters.

6. For each network prefix and scope are temporary addresses are created.

7. Al outbound traffic temporary addresses used on all scopes.

Survey Observations

Although the IPv6 and NDP protocols as outlined by IETF RFCs 4291, 4861, and 4941 were implemented correctly, each implementation had its own unique set of

quirks that were not specifically required by the standards in certain aspects. No two OSs performed the same aftereach of these three tests had been completed.

1. NDP Behavior Test

As per observations, the several instances of host behavior differentiated from the (but not the)recommendations of IETF RFC4861 during the NDP behavior testing. The discrepancy did notaffect the operation of the protocol or the understanding of the NDP packet by the host in eachcase. Each system uses a source address within the Target's subnet (as recommended) when sending unicast or multicast NDP requests, other than the CentOS and Ubuntu Linux hosts, using their link address for all unicast requests (but not multicast Solicitations). In addition, after a failure of a top-layer connection, the Windows Server host also appeared to use its linkaddress as the requester's source.

There were also minor discrepancies in some parts of the packets when sending NDP advertisements. Each advertising package had solicited and Overwrite flags enabled on all systems in the Microsoft Windows family. In response to multicast solicitations, however, boththese flags were enabled by all other systems, only the Solicited flag was enabled when a single application was received. Each Windows system also included in each of its advertises the Target Link-Layer Address option, while others included it only when responding to a multiple-cast request. In addition, every system uses, except for Mac OS X, its link-local address, the address requested by the incoming application as the advertising address.

When processing NDP Requests, Microsoft's Windows Desktop and Windows Server were theonly two systems to behave differently than the rest. Each system added a Source Link-Layer Address to its NDP cache table with an INCOMPLETE or STALE status following receipt of the incoming packet, as specified by RFC. However, except for the above two systems, each system immediately forwarded its application to complete the table entry, even without any real data to be sent to the other system. Each system except Window Desktop and Windows Server accepted the packet, performed its neighbor discovery to look up the other Host's MACaddress, and then transmitted its Advertisement at the newly discovered address when receiving

a Request without the specified source link-layer address option. The wrongly formatted NDPpacket was just ignored on these two systems.

The noted discrepancy did not affect the operation of the protocol or the ability of the receivinghost to understand the NDP packet in each of these cases. In cases where these systems differedin behavior, the words "SHOULD" or "MAY" appeared to be restricted to areas of RFCs where"MUST" was used.

2. Multiple IPv6 Addresses Test

During the tests of multiple IPv6 addresses, several differences between the implementations of the IPv6 protocol of the operating systems were also found. The automatically configured addresses on the Windows Desktop and Windows Server are the first and most significant of these. These two systems have generated a separate, pseudorandom interface identifier for usebased on the calculated address, which is similar to how privacy expands operate, rather than using these auto-configured addresses as the MAC address of the interface (specified in IETF RFC4862 (“IPv6 Stateless Address Autoconfiguration”, 2007)). The alteration identifier persisted through a system reboot, but the fact that this value is permanent or on a log rotation is

unknown. Although this behavior is enabled by default, Windows 7 system can use a "MUST"address. Autoconfiguration procedures are described by the netsh utility in IETF RFC4862. In addition, it has been observed that the Windows desktop sometimes uses its Unique Local- scope address as a source for traffic destined for a link location. This behavior cannot, however,be repeated consistently.

When all systems were sent to traffic, except the CentOS and Ubuntu Linux hosts, their IP address appeared as the packet source that was numerically close to the destination's IP address.Consider, for example, a host with 2001 addresses::/16 and FC00::/8. In the FC00::/8 range, the host address was used as the Source when trying to reach 9999::1. Although FC00:/8 packets could not leave the network of the organization, it was still chosen as the Source because it is numerically close to 9999::1 than an address that starts from 2001::. While no problems with current IPv6

deployments may arise, potential problems might arise when usingaddresses. For example, Microsoft Windows and the MAC OS X hosts may try to reach these addresses using their FC00:: or FE80: addresses when the 8000::/8 or above range has ever been allocated as Global-scene Addresses. Since the addresses correspond to each of the Unique Local and Link-Local scopes, such packages would never reach their destination. CentOS and Ubuntu systems use their global scope to reach all IP addresses except the FC00::/8(Unique Local) and FE80::/16 (Link-Local) areas. The Ubuntu systems use their globalscope. In these cases, the relevant addresses were used in these ranges.

3. Privacy Extensions Test

Whether this feature was enabled or not when the IPv6 protocol was enabled was the biggest difference between systems in implementing privacy extensions for stateless address auto- configuration. Each of the systems tested supported this set of features, but the extensions wereenabled only by default in the Microsoft Windows family. These systems all seemed to haveaset of configurable privacy extensions (which in all cases could be changed by the command- line tools only) and used the default pre- and valid lifetimes of 1 and 7 days respectively according to the IETF RFC4941 recommendation.

4.1.4 Implementation

The details of the implementation of each module in the IPAC system and the configuration ofseveral network administration mechanisms required for the proper operation will be discussedin this section.

Agent Module

The agent process consists of 10 separate parallel threads. The operation of the IPAC Agent system requires a crucial function for every thread. I will explain and describe some concepts of each thread in this section (and one additional class used by the threads). Figure 4.8 illustrates the communication between these threads.

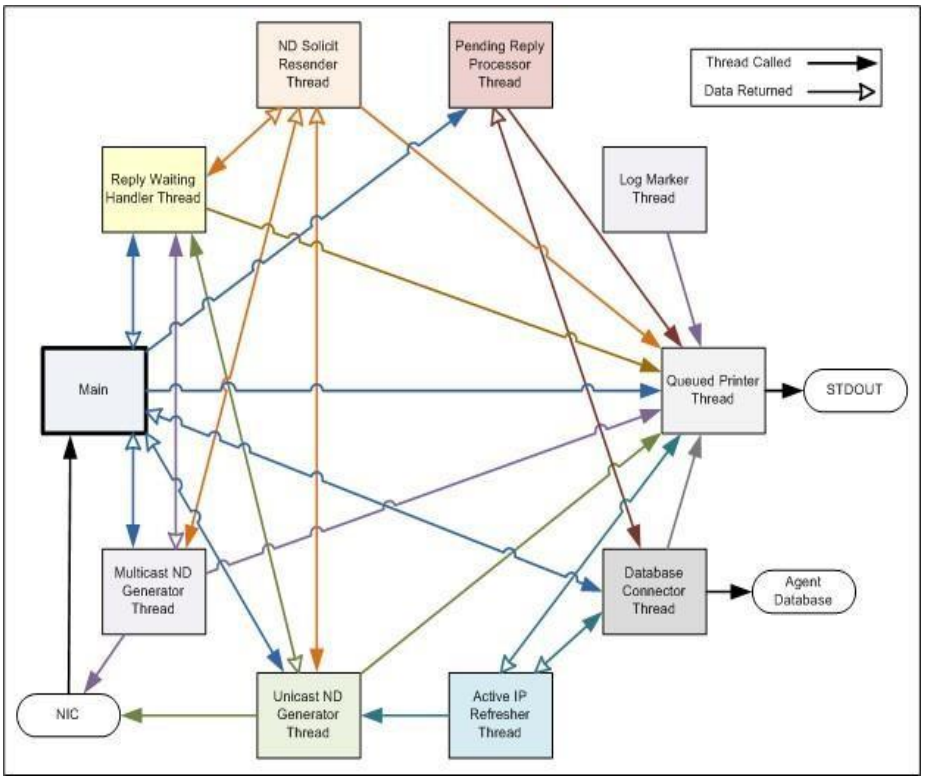


Figure 4.8: Communication between Threads in the IPAC Agent Process

It consists of five different script files for Python. Agent.py contains the main thread body, theipac threads.py contains the bodies of the other threads that are called by the main thread, the ipac daemon.py includes the code used to demonize a host system agent process. The ipac agentconfig.py file is used to customize the agent process. Each of the Agent constant setup, operation, and debugging imported by other scripts are declared in this file. Because the servicecreates a raw listening socket for packets on a host system, this module needs privileged access.

1. Main Thread

The principal-agent thread initializes all threads necessary to operate the IPAC agent. This thread will also create the interface for the packet capture and handle all ICMPv6 packet input.The IPv6 Source, IPv6 Destination, and ICMPv6 Target addresses are processed for each NDPSolicitation or Advertisement packet (if not shipped from the Agent Host), while all other ICMPv6 packet types are discarded.

For each incoming NDP Solicitation packet:

Where the packet comes from the Agent Host, the packet is ignored. The MAC source addressis the system itself. Otherwise, the Source IPv6 address is processed when it is intended to hostthe Aggregate (Destination MAC owns the system) (address processing is discussed below). The Source IPv6 address is processed and the Target IPv6 address is checking if the packet is from or to neither agent host. The remaining packet is ignored if the Target IPv6 address belongs to the agent host, i.e. if the packet is a Multi-Cast Request of the agent. The Target IPv6 address and the destination IPv6 address of the packet are otherwise processed (if it differsfrom the Target and is not a multicast address).

For each incoming NDP Advertisement packet:

The Source MAC address of the packet is checked if the Target Link-Layer Address (TLLA) option is included in the ICMPv6 packet. Any incoherence leads to a warning. The Source MAC address is used as the TLLA, otherwise (the TLLA option is not included). If the packetis from the host agent, the packet is ignored (the MAC source address is the system'saddress). Otherwise, the IPv6 Source and Target addresses are processed using the procedure described in the section 'Source processing,' and a Destination IPv6 address is treated as described in thesection 'Target and Destination processing,' if the packet is not destined for the host agent (Destination MAC is not the system itself). The Agent will then check its list of the Addresseswaiting for Advertisements for the target IPv6 or TLLA addresses (kept

through the reply- waiting thread). When the packet is destined for the Agent host (the Destination MAC is the system's own). If required ads from this IP-MAC address pair, the addresses will be added to the list of the pending advertisements for further processing, which will be maintained by the thread of the Pending Processor. Furthermore, if the IPv6 address of the source differs from theIPv6 address of the target, the source address is treated as described below. If the ad is not expected, the IPv6 Addresses of Source and Target will be processed with the procedure outlined in the following section.

Source Processing:

The system will first check whether there are active IP addresses in the system list for each observed IPv6 source address. If the IP address has not already been activated, a unicast Request shall be sent for verification to the observed IPv6 and MAC source addresses. On theother hand, the associated known MAC address is checked against the observed Source MACaddress if the IP address is already known to have been active. If the MAC addresses

correspond, the Source address will be accepted as true; the IP-MAC pair will receive the sighting record and the IP address of the final sighting is updated in an active list. Otherwise,a multicast Request is sent to verify which addresses remain active on the network (the MAC address does not match whatever have been recorded) any discovered conflicts are handled bythe pending thread of the processor.

Target and Destination Processing:

The system generates a multicast NDP Request for each valid Target and Destination IPv6 address. The system does not try to determine whether the address is already known to be activein contrast to the Source addresses. The reason is that the system should have the same networkview as the host which sent the request. The Agent must always send its own multipacket to ensure that all addresses are known because the solicitude could potentially result in the creation of Unicast Advertisement from more than one host (some may not be known from thesystem).

The program tries to throw the multi-cast requests it sends to prevent the overloading of networks and host resources. When a multicast request is necessary, the system will check its record of Multicast Requests for the objective/destination address recently sent. If the MC_SOLICIT DUPLICATE_INTERVAL address sent in the final second (default: 5), it willnot receive the new solicitation. In some situations, such as when the system observes a host constantly requesting a non-existent IPv6 address, this prevents external multicast traffic generation. This could, however, lead to a scenario where an address remains invisible for a certain period. If a host uses the address specified in the time window of the MC_SOLICIT_DUPLICATE _INTERVAl, the system will not confirm this until it appears inthe NDP Request Source address or, after a threshold window passing, when it is seen as the Request Target or Destination. The first should take place on most systems as soon as the system has data it needs to send from that address, as shown in the IPv6 Implementation Surveyresults.

ReplyWaiting Thread

A list of IP and MAC addresses from which the agent expects to receive an NDP Ad is kept inthe ReplyWaiting thread. This list is used to differentiate between incoming NDP node advertising from IPAC generated requests and other unrequested advertising. Any publicity notexpected to be valid to prevent the poisoning of the agent's records. Although several

threads have to recover data from this list and change the content, the ReplyWaiting thread has direct access only. Additional threads can access the list by several commands.

The response-waiting thread lists and sequentially executes incoming access/modification commands to prevent any data integrity problems that may occur when the list is accessed or modified by more than one thread simultaneously.

The ReplyWaiting list includes a dictionary containing the key to an IP and MAC address pair, and a timestamp, response count, timeout count. The time-stamp identifies the number of advertisements received from an IP/MAC pair after the last request was received, and the time-out count records the number of times the request was received after no advisory has been received within the time-out threshold. The time-stamp indicates the last time-out pair was sent.When multicast requests are sent,

the MAC address is entered into 'NULL,' and the system checks the list of the IP-MAC and IP-'NULL pairs on the incoming Advertises.

Additional threads interact with the answer waiting and its internal list by calling the action function of the thread and passing a string (optionally with additional parameters for the command). The action function adds to an internal queue the command and all options. If the command is processed by the body of the ReplyWaiting Thread, any results are returned to thefunction via the Queue results. The commands and outcomes are tracked and arranged by a time signature value created when the action function is called to prevent results from being returned to the wrong thread when multiple threads interface with the ReplyWaiting thread concurrently.

The body of the ReplyWaiting thread monitors the incoming command queue and serially processes the commands. The command 'Add' causes a pair IP/MAC to be added into the list and the 'get' command returns the values for the given IP/MAC pair, the 'get' returns the entirelist of contents, and the 'hake' command checks the list for that IP/MAC pair. The 'Hash' command returns the IP/MAC pair for that pair. When publicity for the agent system is observed, "incomingAdv" is used by the main thread. The thread checks the list of IP/MAC and IP/NULL pairs with this command. If any is detected, it is tagged to see whether the entryis within the age threshold of SOLICIT TIMEOUT INTERVAL. If it did not have a timing, the answer counter will be increased to indicate that an advertisement is expected byreturningthe value of 'True' in the main thread. The command 'solResend' is used to identify IP / MAC address pairs without receiving any ads through the Solicitation Resender Thread. The thread travels the list with this command and checks for each entry. If the entry is a match, a list is

added that is returned to the calling thread by adding an IP/MAC address pair. If the maximumretrieval of the entry is reached, it will be deleted from the list by the name SOLICIT RETRY COUNT.

PendingProcessor Thread

A list of IP/MAC address pairs from which an NDP ads have recently been received from the main agent thread is held in the PendingProcessor Thread. The agent can recognize scenarios with multiple MAC addresses claiming to be owning the same

IP address from this list of pending addresses. The IP/MAC address pairs added to the database sighting registrations andActive IPs lists will be available only after the passage of the SOLICIT_TIMEOUT_INTERVAL.

The Pending Processor thread is informed by adding the function when the main thread receives an expected NDP advertisement. This function adds the IP, MAC, and timeline values of the main theme to the queue. This queue is periodically checked and added by the group of the Pending Processor thread to its internal list. The thread will then iterate the list and examine each IP and MAC address, once the list has been updated. A record will be created and the Active IP list will be updated with the new timestamp and the retry zero (or added when the IP/MAC does not exist already) for each IP address that has achieved a SOLICIT TIMEOUT INTERVAL if it only has one associated Mac address. If the IP has more than one MAC address, a record for each MAC has been created, both of which are added and updated to the Active IP list, and a warning about the problem has been raised.

SolicitationResender Thread

The retransmission thread of the NDP Requests into IP addresses for which NDP Advertising is expected but not received within the time-out. In cases when the request or advertising packetdid not reach its destination due to link congestion, this reduces the chances of the agent processfailing an IP address.

The body of the Solicitation Resender thread regularly requests the Reply Waiting thread for a list of IP/MAC address pairs that have not received replies in the time period. For every IP/MAC pair, the ND Unicast Generator or (when the mac is 'NULL') ND Multicast Generator threads are directed by Solicitation Resender to generate a new request for the target IP addressfor each IP/MAC pair.

ActiveIPRefresher Thread

The ActiveIPRefresher thread allows the agent system to verify that an IP/MAC address pair that has previously been observed remains active in the network even if NDP traffic is not generated continually. If during the last address aging interval, the agency does not observe any traffic from the IP/MAC pair, it produces a unicast Solicitation, which causes publicity toreturn from the goal.

In the 'active' table not displayed in the last ACTIVE IP REFRESH INTERVAL, the ActiveIPRefresher thread periodically queries the agent database for IP/MAC address pairs viathe data interfaces thread. The pair is taken away by the "active" database table and a record isgenerated to mark its closing, if an IP-MAC pair has reached the ACTIVE IP REFRESH COUNT threshold without any Advertises being returned to the agent. Otherwise, a unicast solicitation will be generated and the associated input in the 'active' database table updated to reflect the increased retry count.

NDMulticast Generator Thread

The NDMulticastGenerator thread processes all NDP multicast solicitations sent by the Agent.Other threads interact with the thread by creating a function that accepts a parameter of the IPaddress of the goal. This function adds an internal queue address, which the body of the threadmonitors continuously. The ICMPv6 NDP Solicitation packet will be created for each addressusing the Packet Constructor class, the packet injector on the network interface of the host willbe set and the manufactured packet will be injected into the Network. The thread adds the targetto the Reply Waiting list via the Reply Waiting Thread before the packet is sent (it is done in this order to prevent the resulting Advertisement from arriving before it is expected). The threaded body then creates a timestamp that marks when the packet is sent and adds it to the internal result queue to return the value to the calling thread by the generating function.

NDUnicastGenerator Thread

The NDUnicastGenerator thread processes all NDP unicast solicitations packets sent by the agent. Other threads interact with this thread via the generating function that accepts the parameter IP and MAC of the target. This feature adds the addresses to an internal queue that is monitored continuously by the thread corpus. The ICMPv6 NDP solicitation packet is createdfor each address using the Packet Constructor class, a packet injection socket is installed onto the system's network interface, and the created packet is injected into the network. The thread addsthe target to the Reply Waiting list via the ReplyWaiting Thread before the packet is sent (it isdone in this order to prevent the resulting Advertisement from arriving before it is

expected). The threaded body then creates a timestamp that marks when the packet is sent andadds it to the internal result queue to return the value to the calling thread by the generating function.

Database Interface Thread

A unique interface to the MySQL database of the agent is the Database Interface thread. Whileseveral threads need to obtain and modify records in the database, Database Interface threads perform all queries and updates to the database tables. This prevents problems of data integrity that could occur when multiple threads simultaneously try to read or write.

When initialized, a connection to the Agent database is provided by the Database Interface thread. If FRESH_START Trial mode is enabled, all existing records in new data preparation will be deleted from the 'active' and 'records' database tables. Additional threads interact with the execution function with the Database Interface thread. This function contains a text string with the parameter of a set of SQL commands and adds the item to the inner queue.

The Database Interface thread body monitors the queue for input commands continuously. Oncea new item is received, SQL will be executed, data from the database will be obtained, changeswill be committed and all results will be added to an internal results queue. A running functioncan access the results queue, which returns the results to the calling thread. If there is an exception when the SQL commands are executed (such as a primary key duplicate), the exception is taken and a vacuum setting is instead returned.

Queued Printer Thread

The Queued Printer thread serves as a unique interface to write messages in the output and errorlog of the agent. When multiple parallel threads use the same text output file, and very likely, try to print text simultaneously for 2 or more. When this happens, this text is interwoven and produces unreadable text. The Queued Printer thread resolves this problem by queuing messages to be attached one by one to the output log.

Another thread transmits text via the printq function to the Queued Printer. These function parameters the message text (and optionally the name of the originating

thread for debugging purposes), adding a tuple of these values plus an internal queue time stamp. The Queued Printerbody continually monitors and prints the queue for incoming messages. The name of the originating thread will also be added to the output file if the PRINT_DEBUGGING mode is activated.

Log Marker Thread

In the Agent's output and error log, the LogMarker thread produces regular MARK messages.These messages inform the user that there are no other messages within the log if the agent process is still running. These messages include, in addition to the current time, counts for NDPrequests and advertisements the agent has processed since the last MARK. These log messagesare generated by the Queued Printer Thread on the Agent Log once every LOG_MARK_INTERVAL (default: 15 minutes).

Packet Constructor Class

The Class Packet Constructor includes a set of functions that assemble the NDP Request packetsfor injection into the network. The NDUnicast Generator and ND Multicast Generator threads are respectively used for unicast NDSol and multicast NDSol functions. The other two calls fora third function, completePkt, to finalize the raw Ethernet framework.

The function unicast NDSol takes parameters for sources and target IP and MAC, converting the values in ASCII to binary for insertion into the raw packet data. The ICMPv6 portion of the packet is created using the Python dpkt module using the Target IP as the target value andthe Source MAC as the Source Link-Layer Address value. The fullPkt function is called to build the rest of the packet and the whole packet is returned to the calling thread.

The Multicast NDSol function uses the parameters of source IP, target IP, and source MAC, which are converted into binary form to be used in raw packet data. Based on the given target IP address, Multicast Destination IP and MAC addresses are developed. With the help of the Python dpkt module, the ICMPv6 part of the packet will be created by using the Target IP as the Target value and the MAC as the Source Link-Layer Address value. The fullPkt function is called to build the rest of the packet and the whole packet is returned to the calling thread.

The full Pkt function takes ICMPv6 as a parameter, as well as the Source and Destination IP and MAC address binary form, generated by the Unicast NDSol or Multicast NDSol datagrams.Using the dpkt Python module, the ICMPv6 datagrams can be used to build an iPv6 packet andan Ethernet frame for payload use with the IPv6 packet. The finished Ethernet framework is returned to the calling function, which returns it for injection to the network to the original thread.

4.1.5 Trap Handler Module

A single thread that constantly lists SNMP trap messages consists of the Trap Handler process.This module has 2 Python script files, the process body is included by ipac traphandler.py, andthe host system is used by ipac daemon.py for the demonization process of the Trap Handler. The process is adapted using constants initialized at the start of the ipac traphandler.py script. The values of these constants determine the initial actions of the Trap Handler, the database itstores, and its debugging. As the service creates a socket within the limited range of UDP ports,this module requires privileged access.

This process begins by setting a UDP port 162 listening socket for SNMP traps and notifications as the default destination. The PySNMP Python module offers socket functionality. The socket will be processed individually for all SNMP packets received, as wellas each Variable Binding pair in the packet. In the 'mac notifications table" of the Trap Handlerdatabase, a record is established for the variable binding pair, where either a MAC Learned ora MAC Removed notification is contained.

Two special modes for testing and debugging Trap Handler functionality are provided by this module: FRESH START and TEST MODE. When FRESH START mode is activated, when it is initialized, the Trap Handler process will purge any existing records from its database. Foreach testing session, this provides a clean slate and the Trap Handler works as usual. On the other hand, TEST MODE processes every trap notification as normal, while preventing any change to the database of the module. This enables us to debug the operation of the module without affecting existing data.

4.1.6 Server Module

The server process consists of one thread, which regularly resumes and cleanses records from the Agent and Trap Handler processes. This module consists of 4 Python script files separately. Ipac_server.py contains an agent's recording code body. Ipac_trapsum.py includes the Trap Handler recording code, the host server processing code is demonized by ipac_daemon.py and a set of server module constants is included with ipac_server_config.py. These values determine the initial actions of the server, its processing behavior, the database used for the storage of SNMP trap logs, and its debugging activities. The term "session" is used in this section for the time interval known to exist in the network as the IP and MAC address pair (IP-MAC) or MAC and the physical switch pair (MAC-Port).

The server starts from its database with the list of known agents. It will query the relevant Agent database for the IP-MAC viewing records of each agent that has been marked active. Upon obtaining these records, they are removed from the agent database to prevent reprocessing of old information by following server process iterations. It then passes through the list of records received (sorted by the IPv6 address and then timestamp), aimed at reducingto one record a multitude of viewing records for the same IP and MAC address session, indicating the first and the last time the pair has been viewed. Two adjacent sighting agent records are classified as part of the same session if they contain the same values for IP and MAC and the time stamp delta is not bigger than the permitted timeout value. The session is considered to be finished if a termination record is seen or a timeout between documents is reached. In this case, the server database creates a summary record, and processing the new session begins with the upcoming agent record again. An ending type value indicating whethera Mac-IP binding session is considered terminated or continuous is applied to each record created by a Server. A final type value of 20 indicates that there was a termination record and the pair left the network, and 10 indicates that the pair was still active when the agent last updated their database. The server is considered to be a permanent session and records are generated in the server's end-type-value database if it is used by all the entries having the same IP and MAC addresses without viewing a definite termination record.

Despite its violation of the standards in the protocol, the same IPv6 address may be viewed simultaneously with multiple MAC addresses. The server identifies instances

by seeingsuccessive data with the same IPv6 address, various MAC addresses, and timestamp delta below the timeout value. In such cases, each contradictory pair receiving its summary document shall be treated individually. However, the records identified for each of these pairsare linked to the results obtained from the Agent Database and make the actual beginning andconclusion of each session difficult to identify. To solve this, the server will move the second pair of records to a secondary list every time a conflict is identified and temporarily ignore them. The original pair records are processed as normal. Using the same procedure forsessionrestructuring and conflict identification, when all remaining records have been processed (andany other conflicts solved), the server will return to the secondary list. Until all conflicting pairsare processed, the server continues to iterate via this list.

Upon processing of all the agent records, Trap Handler records are processed using the same procedures. However, only when the MAC-Port pair is first and last seen (instead of every 30seconds with agent records) are the records generated by Trap Handler, each record is

processed individually. The Server determines whether an instance is indicated when the Macaddress was retrieved from the switch or removed from the switch for every record received from the Trap Handler database. If the server is a 'Removed' record, it examines its database to see whether there is a matching unclosed MAC-Port session. If it is, the record will be updatedand closed with the appropriate end time; otherwise, a new record is created and closed. Whenit's 'Learned,' the server checks its base again to see whether the MAC and port have an unclosed session. If there is no unclosed record, a new record will be created and closed (without an ending time). Otherwise, before a new record is created, the existing record is finished with an error code (as no deleted record has been received).

The server procedure implements a 15-minute waiting period between each log acquisition andprocessing process to prevent overload of the Server host or the Agent databases. This moduleprovides, just like the Trap Handler, two specific modes to test and debug it: FRESH_START and TEST_MODE. Just like in the past, FRESH_START mode allows the server to remove from its database all existing binding and TrapBinding documents, leaving a fresh testenvironment. But TEST MODE works differently on the server. It will prevent the deletion ofrecords that

received from the Agent and Trap Handler databases, rather than preventing records from being written to the database.

This allows for the reprocessing of records during each iteration on the server, so that the agentand Trap Handler databases cannot be repopulated between testing sessions.

4.1.7 Interface Scripts

The final module includes a set of scripts that provide an interface for an administrator or investigator to access IPAC data. These four displays each give a different view of the stored data: one gives a tool to query the system's bright summary records, the other uses a toil to view an IPv6 addressing timeline visualization, and the last two processes text-based application logs and binary packet capture. Within the output of each script the IPV4 ID, port,and VLAN ID (e.g. '192.168.0.20/0003/0010' would be VLAN 10 on port 3 on the IPv4- address switch of 192.168.0.20) were included with every physical switch port identified.

Database Query Interface

A script for ipac_query.py provides several ways to obtain certain information through a command-line utility from the IPAC server database. This tool can perform five different operations: list all the IPv6 addresses in the network, list all of the network's MAC addresses, list all of the physical switch ports in use with the IPv6 or MAC address specified, list all IPv6

and/or MAC addresses in use on the network, or list all instances where IPv6-MAC address conflict has occurred. These operations may be limited to one point, starting and/or ending times, except the listing of currently active addresses (matches records ending after and beginning before the specified times, respectively). Moreover, the operation of all IPv6addresses on the network can be restricted to those paired with a specific MAC address and allMAC addresses on the network can be restricted to those paired with an IPv6 address.

When running this script from a terminal, the user specifies operations and limiting parameters.The use guidance provided in Figure 4.9 shows these options and an example of their output can be found in Figure 4.10.

Figure 4.9: ipac_query.py Usage Guide

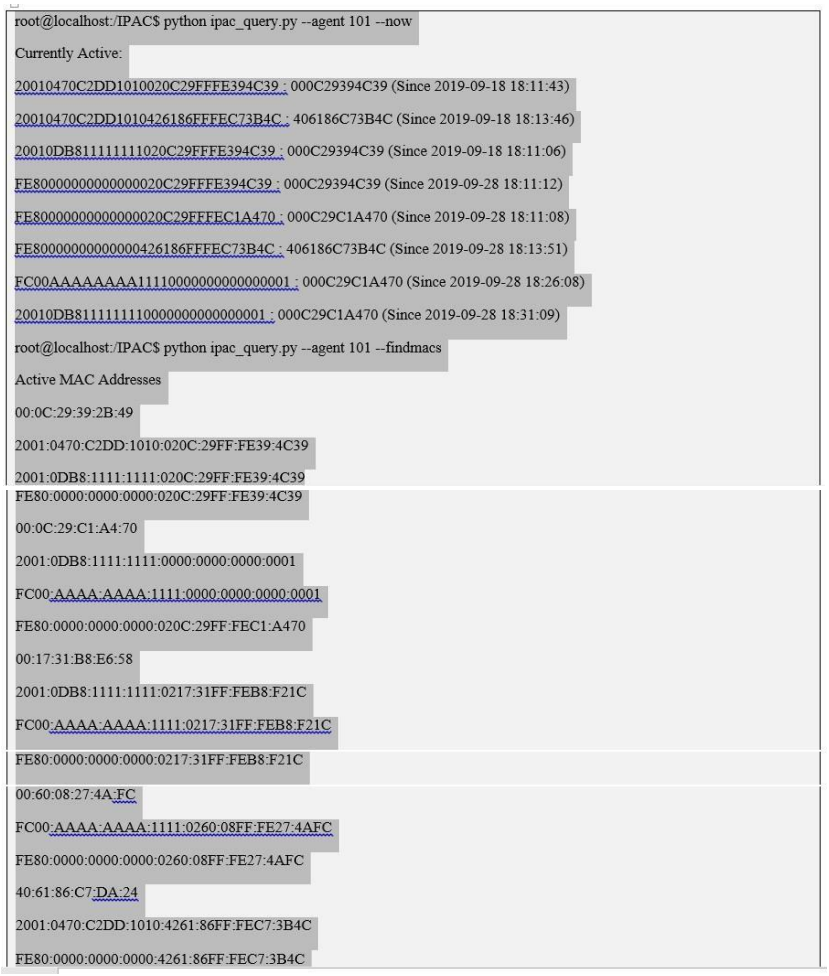


Figure 4.10: Example ipac_query.py script output

Interactive Timeline Graph

This ipac_graph.py is an interactive visual timeline for the IPv6 address use of the network of network operators. This dynamic graph represents a separate timeline segment of each IP-MACsession derived from the IPAC Server database based on the BrokenBarH graph provided by the Matplotlib Python module. Each IPv6 address in the network is indicated by the Y-axis, whereas the X-axis is time. Each section is marked with the MAC address for the session and a red or green tab that shows whether the session has ended (red) (green). All IP-MAC conflictsare identified by a red highlighting bar and an orange starburst (superposed segments). The graph is interactive because it allows the user to zoom in and display the information from various time scales. It also allows the user to click on a segment to show the exact start and endtimes of the session. Figure 8.9 in Section 8.3 gives a screenshot of this graph.

Text Log Processing

The ipac textlog.py offers a tool to process IPv6-contact text-based log files automated. The IPv6 addresses that are identified in the log file shall be replaced in this scribble at the time identified by the log entries (if available in IPAC Server database records), by information

about the MAC addresses and switch ports related to the address. The script ipac_textmog.py is used to find IPv6 and timestamps in any log entry using regular expressions. While it can find IPv6 addresses in any log format, it is only currently able to identify timestamps in files using Syslog format, as well as Apache access and error logs; additional log formats can be adapted to fit timestamp structures by adding the proper regular expression.

After finding the correct IPv6 and timestamp addresses in each log entry, the script will decidewhich agent it is to ask for the right MAC address information. The agent of the network the Logfile originated form is used for all Link-Local scope addresses (requested from the user when the script initializes). It uses the list of the known subnetwork in the IPAC server database to identify the appropriate agent to be consulted for all other address types. If the address doesnot match any of the server's subnet prefixes, the script will never know which records are to be searched and will not try further address processing.

Once identified the associated MAC addresses and physical switching port, the log entry and output will be added to a separate file, encapsulated with brackets <[and]>, to delimit this information from the original content of the log. The system identification time frame as mentioned marks each MAC address and port ID to show confidence in the identifier with a succession of asterisks or care. This reliability assessment is based on whether a duplicate wasfound (more than a MAC/port for the requested IP/MAC) and if the result was exact or ±30 second (referred to as a "close" match) window within the specified timestamp. Figure 4.11 explains the significance of these sequences, and Figures 4.19 and 4.20 show an example of the output of the script.

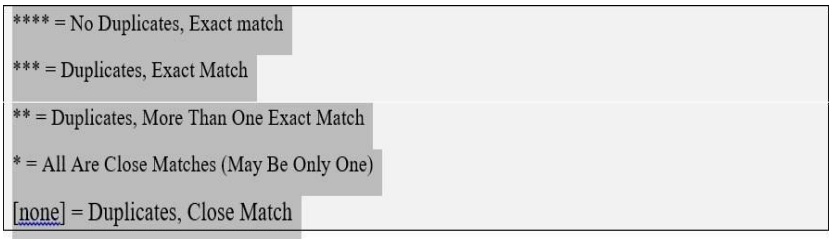


Figure 4.11: Confidence Indicators for MAC addresses in Text Logs Reports.

(Values also apply to the '^' marks for Switch Port identifiers)

Libpcap File Processing

The final Interface script, ipac_pcaplog.py, provides a function similar to the previous text logprocessor, but for binary packet capture files in the Libpcap format. This utility utilizes the dpkt Python module to disassemble and processes each packet in the provided capture file. Foreach IPv6 packet in the file, the script will attempt to correlate its Source and Destination IPv6

addresses (except any Loopback or Multicast addresses) to a MAC address and physical switchport by querying the Server database. In the current version of the script, no attempt is made toidentify any IPv6 addresses existing beyond the IP header. Just as with the previous interface script, determining which Agent records to query is accomplished by examining the list of known subnets in the Server database. Rather than appending to the packet capture file, all summaries are written to a separate text report and identified by the number of the corresponding packet in the original file. An example of the script's report output can be seen in Figure 4.22.

Databases

For record storage and data processing, the IPAC system uses three different types of databases.The Agent module host uses a database to store IP address information that is currently activeon its network and to record historical information on each instance of an IP address in NDP traffic. Trap handlers use a database to save every notification from an Ethernet switch that waslearned or removed from the MAC. The server module uses a database that includes IPAC andagent and traps handler records summaries. Every host uses its local MySQL server in the demonstration network. Further the structure and implementation of these databases.

Time Synchronization

The IPAC Server host, which maintains its Clock synchronized to Stratum 1 or Stratum 2 time source at pool.ntp.org, provides central NTP service in the demonstration network. Figures 4.11, 4.12, and 4.13 provide examples of NTP configuration, and NTP commands for ntpd, ntpd, and Cisco IOS client.

```
driftfile /var/lib/ntp/ntp.drift
server pool.ntp.org
restrict -4 default kod notrap nomodify nopeer
restrict -6 default kod notrap nomodify nopeer
restrict 127.0.0.1
restrict ::1
restrict 10.100.111.0 mask 255.255.255.0
```

Figure 4.11: Sample Linux NTP Server Configuration File

```
driftfile /var/lib/ntp/ntp.drift
server 2001:db8:1111:1111::2
restrict -4 default kod notrap nomodify nopeer noquery
restrict -6 default kod notrap nomodify nopeer noquery
restrict 127.0.0.1
restrict ::1
restrict 10.100.111.114
```

Figure 4.12: Sample Linux NTP Client Configuration File

```
clock timezone IST +5.30
clock summer-time IST date Mar 23 2019 2:00 Jan 6 2021 2:00
ntp peer 10.100.111.114 prefer
```

Figure 4.13: Sample Cisco IOS NTP Client Configuration Commands

Use of IPv4 and IPv6

For most internode communications, the IPAC system and the demonstration network use theIPv6 protocol. However, two parts of the system require the IPv4 protocol to be used. IPv4 must first be connected to the respective databases of the scripts. While the MySQL servers onthis network support remote connectivity over IPv6, current versions of MySQLdb Python canonly link to IPv4 addresses. The only

way to access the databases is to connect to the IPv4 server address without any heavy modification of the MySQLdb module.

The other area currently required for use in IPv4 is the treatment of SNMP traps. While the SNMP protocol can fully transport the traps and notices via IPv6 and the PySNMP Python module can listen to IPv6 notifications, the Ethernet card does not support IPv6. Many companies still use older hardware in their network infrastructure are probably also confrontedwith this issue. Until Catalyst 2960 and 3560 Ethernet family releases of IOS-released IOS switches 12.0(22) or 12.2.(2) were released by the Cisco Switching Hardware (actual IOS versions providing IPv6 functionality vary between release branches). Where the network hardware does not support IPv6 management capabilities, the NTP clock synchronization mechanism should also operate via IPv4.

MAC Notifications

Each Ethernet Access Layer Switch on the organization's network needs to be configured to provide the IPAC System with information on each of the MAC addresses it knows to detect which physical ports each MAC address uses. The mechanism that the IPAC uses for this purpose is SNMP CAM events, but in this demonstration system, its recommended to focusedonly on Cisco's Catalyst series of IOS 12.0 switches and later. This feature provides the Enterprise Ethernet switches from a variety of different vendors. Figure 7.8 contains a sample

configuration of the commands to allow these notifications of MAC address on a Cisco Catalyst2950 switch running IOS release 12.1(22)EA14.

```
!--- Enable MAC Notification on all non-uplink/non-trunk ports
interface FastEthernet0/1
name Uplink
!
interface FastEthernet0/2
snmp trap mac-notification added
snmp trap mac-notification removed
!
interface Vlan1
ip address 10.100.111.12 255.255.255.0
no ip route-cache
!
snmp-server community public RO
snmp-server enable traps MAC-Notification
snmp-server host 10.100.111.114 public
!
!--- Send notifications immediately after event
mac-address-table notification interval 0
mac-address-table notification
!--- Inactive MACs removed from CAM table after 60 seconds
mac-address-table aging-time 60
end
```

Figure 4.14: Sample SNMP Notification Configuration for a Cisco Catalyst 2950 Switch.

It is important to ensure that the MAC notifications are only configured for switch ports that have direct access to end devices. The switch will inform the IPAC system of MACaddressedwith other switches if the uplink port is set up to send these updates, resulting in duplicate messages. Since only looking for the physical port to which the device is directly connected, the accuracy of the system could be affected by messages from these uplink ports. In cases where a port connecting to a different switch needs to be configured to send such notifications,however. If the second unit cannot send its updates to the CAM table to the IPAC system, that

configuration would at least provide some network status visibility. Although these scenarios do not allow an accurate physical segment to be identified to which a node was connected, thesystem may at least reduce the scope of the connected switch.

4.1.8 Results

In a platform that seeks to replicate a corporate network environment at a small scale, it has been demonstrated the capacity and function of the final IPAC system. In this

section, the discussion held on the network and methods used to demonstrate this system and provide examples of the use of its IPv6 address tracking functions for administrators and researchers.

Demonstration Network

The network used to demonstrate the IPAC system operation sought to reproduce on a small scale what a company could do in one office. There were three sub-networks on the network: one with host systems ("LAN A"), a third with server systems ("LAN B"), and a third networkconnecting external networking ("WAN"). Each network was allocated a separate subnet of theGlobal Scope (in 2001:/16), and the two internal networks also used a Unique-Local Scope (inFC00:/8). Both networks were deemed to have their own IPAC agents, each with one IPAC server handling the agent records and SNMP trap messages from the Ethernet switches. On theLAN B and WAN portions of the network, web servers for use in test traffic generation were also provided. Figure 8.1 shows this network topology.

Ubuntu desktop operating systems with Python 2.x and MySQL Server have implemented theIPAC Agents and Servers. Cent OS hosts with Apache-provided web servers. A host used theVyatta Core 6.0 routing platform to route between the networks. Several Windows Professionalsystems and Ubuntu systems were added to LAN A.

This demonstration network relied heavily on routing and network services through virtualization. A single server running the 64-bit Ubuntu Desktop LTS as its host operating system provided all the virtualization systems. Both the IPAC agent and the IPAC server existed as an image for the Oracle VirtualBox, while VMware Workstation was used for the web servers and routers. The behavior observed during the IPAC system testing phase requiredthe use of both virtual software packages. Initially completely deployed within the VMware workstation environment, after observing instances in which VMware did not properly forwardICMPv6 traffic to the virtual devices, IPAC Agents and Server have migrated to VirtualBox images. After determining that their operation would not be affected by this behavior, the other

virtualized services were laid on the first platform. In the dashing blue line of Figure 4.15 the systems are contained in these two virtualized environments (on the same physical host).

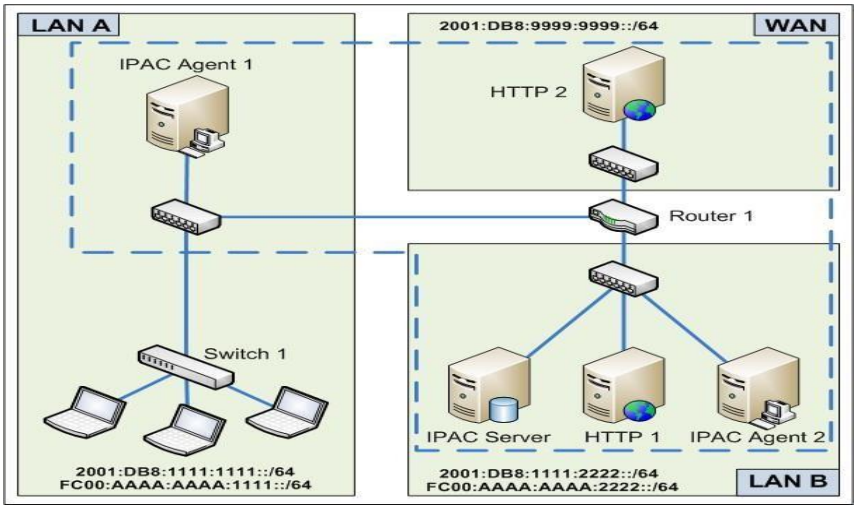


Figure 4.15: Demonstration Network Topology

NDP Traffic Processing / IPAC System Operation

The proof of IPAC was started with the agents, servers, and network infrastructure, but host nodes were separated from the network. These host nodes werethen connected to the Ethernetswitches individually. After every physical connection was created the switches gave the TrapHandler module an SNMP notification that they had learned about the MAC address of the newhost. When the messages were received, a notification MAC record was created as shown in figure 4.16 by the Trap Handler process.

```
mysql> select * from mac_notifications;
+---------------+--------------+------+------+-----------+------------------+
| switchIPv4 | macAddress | port | vlan | operation | timestamp |
+---------------+--------------+------+------+-----------+------------------+
| 10.100.111.21 | 006008274afc | 000a | 0001 | 1 | 1304029177.94567 |
+---------------+--------------+------+------+-----------+------------------+
1 row in set (0.00 sec)
```

Figure 4.16: Sample SNMP trap record obtained from the IPAC Trap Handler database

Every host would then connect to the web servers on the LAN B and WAN portions of the network after the switch is connected. As part of this process, NDP traffic

between those nodesand the router has been exchanged containing the unique local and global addresses of the host.After observing that traffic, the Agent checked the existence of the addresses, added them to

its list of known active IP addresses, and created a sighting record that marks the time of the observation of the addresses. Examples are shown in Figure 4.17 of these records.

```
mysql> select * from active;
+----------------------------------+--------------+-------------------+----------------+
| ipv6Address | macAddress | lastSeen | retryFailures |
+----------------------------------+--------------+-------------------+----------------+
| 20010DB811111111020C29FFFE394C39 | 000C29394C39 | 1304029615.33424 | 0 |
| 20010DB81111111110000000000000001 | 000C29C1A470 | 1304029603.19496 | 1 |
| FC00AAAAAAAA11110000000000000001 | 000C29C1A470 | 1304029627.85339 | 0 |
| FE80000000000000020C29FFFEC1A470 | 000C29C1A470 | 1304029612.45889 | 0 |
| FE80000000000000020C29FFFE394C39 | 000C29394C39 | 1304029620.5273 | 0 |
+----------------------------------+--------------+-------------------+----------------+
5 rows in set (0.00 sec)

mysql> select * from records;
+----------------------------------+--------------+-------------------+------+
| ipv6Address | macAddress | timestamp | type |
+----------------------------------+--------------+-------------------+------+
| 20010DB81111111110000000000000001 | 000C29C1A470 | 1304029603.19494 | 11 |
| 20010DB811111111020C29FFFE394C39 | 000C29394C39 | 1304029578.41214 | 11 |
| 20010DB811111111020C29FFFE394C39 | 000C29394C39 | 1304029615.33424 | 11 |
| FE80000000000000020C29FFFE394C39 | 000C29394C39 | 1304029578.53811 | 11 |
| FE80000000000000020C29FFFE394C39 | 000C29394C39 | 1304029620.52726 | 10 |
| FE80000000000000020C29FFFE394C39 | 000C29394C39 | 1304029615.46038 | 11 |
| FE80000000000000020C29FFFEC1A470 | 000C29C1A470 | 1304029599.39645 | 10 |
| FE80000000000000020C29FFFEC1A470 | 000C29C1A470 | 1304029609.39881 | 10 |
| FE80000000000000020C29FFFEC1A470 | 000C29C1A470 | 1304029612.45889 | 10 |
| FC00AAAAAAAA11110000000000000001 | 000C29C1A470 | 1304029590.93067 | 11 |
| FC00AAAAAAAA11110000000000000001 | 000C29C1A470 | 1304029627.85339 | 11 |
+----------------------------------+--------------+-------------------+------+
11 rows in set (0.00 sec)
```

Figure 4.17: Sample Active IPs (top) and Address Sighting (bottom) record obtained from IPAC Agent 1 (LAN A)database.

The IPAC Server would obtain the sighting of addresses and SNMP trap documents from the Agents and Trap Handler databases at regular intervals. These records have been processed togenerate IP-MAC and MAC-Port binding logs. Figure 4.18 shows examples of the records generated and saved by the IPAC server.

```
mysql> select * from bindings;
+----------------------------------+--------------+---------+------------------+-
| ipv6Address | macAddress | agentID | startTime |
+----------------------------------+--------------+---------+------------------+-
| 20010DB81111111110000000000000001 | 000C29C1A470 | 101 | 1304029269.25769 |
| 20010DB811111111020C29FFFE394C39 | 000C29394C39 | 101 | 1304028066.85902 |
| 20010DB811111111021731FFFEB8F21C | 001731B8F21C | 101 | 1304028759.19376 |
| FE80000000000000020C29FFFE394C39 | 000C29394C39 | 101 | 1304028072.17825 |
| FE80000000000000020C29FFFE7A32D5 | 000C297A32D5 | 102 | 1304028314.85705 |
| FE80000000000000020C29FFFEC1A470 | 000C29C1A470 | 101 | 1304028068.39946 |
| FE80000000000000021731FFFEB8F21C | 001731B8F21C | 101 | 1304028721.54399 |
| FE80000000000000026008FFFE274AFC | 006008274AFC | 101 | 1304029183.0272 |
| FE800000000000000A0027FFFE0D760A | 0800270D760A | 102 | 1304028105.68617 |
| FC00AAAAAAAA11111000000000000001 | 000C29C1A470 | 101 | 1304028968.96334 |
| FC00AAAAAAAA1111021731FFFEB8F21C | 001731B8F21C | 101 | 1304028716.41991 |
| FC00AAAAAAAA1111026008FFFE274AFC | 006008274AFC | 101 | 1304029177.90177 |
| FC00AAAAAAAA2222020C29FFFE7A32D5 | 000C297A32D5 | 102 | 1304028309.47606 |
| FC00AAAAAAAA22220A0027FFFE0D760A | 0800270D760A | 102 | 1304028100.4984 |
+----------------------------------+--------------+---------+------------------+-
```

```
-+------------------+---------+
| endTime | endType |
-+------------------+---------+
| 1304029566.2727 | 10 |
| 1304029541.36319 | 10 |
| 1304029342.34651 | 20 |
| 1304029583.6042 | 10 |
| 1304029598.88032 | 10 |
| 1304029573.15098 | 10 |
| 1304029347.09259 | 20 |
| 1304029447.35751 | 20 |
| 1304029619.06309 | 10 |
| 1304029553.75552 | 10 |
| 1304029342.09427 | 20 |
| 1304029454.62616 | 20 |
| 1304029593.65372 | 10 |
| 1304029593.40076 | 10 |
-+------------------+---------+
14 rows in set (0.00 sec)
```

```
-+------------------+---------+
| endTime | endType |
-+------------------+---------+
| 1304028638.2214 | 20 |
| 1304029352.12969 | 20 |
| 1304029488.71055 | 20 |
-+------------------+---------+
3 rows in set (0.00 sec)
```

```
mysql> select * from portBindings;
+--------------+---------------+------------+------+------------------+-
| macAddress | switchID | switchPort | vlan | startTime |
+--------------+---------------+------------+------+------------------+-
| 001731b8F21C | 10.100.111.21 | 0007 | 0001 | 1304028538.82958 |
| 001731b8F21C | 10.100.111.21 | 0007 | 0001 | 1304028716.60959 |
| 006008274afe | 10.100.111.21 | 000a | 0001 | 1304029177.94567 |
+--------------+---------------+------------+------+------------------+-
```

Figure 4.18: Sample IPv6-MAC (top) and MAC-Port (bottom) summary records obtained from the IPAC Server database.

Interface Scripts Operation

A series of application logs and packet captures have been created to show scripts of IPAC interface during the IPAC system operation demonstration. These include an HTTP Server Apache access log on LAN B, a central Router iptables log (via Syslog), and short LAN A packet capture.

Apache web server and iptables were able to manage the log files from the ipac_textlog.py script. This script has added a report on the usage of IPv6 addresses at the end of each log entry,

as shown in figures 4.19 and 4.20. This report, included in the "<[" and "]>" brackets, covers every IPv6 address in the log entry, any IP-related MAC addresses, and any MAC-related switch ports. In case an IPv6 does not have a corresponding MAC address or switch port, the text has either been replaced by 'Unknown' or 'Unknown Ports.' Furthermore, the confidence rate in one to four asterisks (mac addresses), or carats (switch ports), was marked on each MACaddress and switch port ID. In cases of multiple IPv6 addresses found in a log entry, "||" is theseparation of each address report. In the following format, the report was structured:

<[IPv6 Address -> MAC Address*Confidence (Switch ID/Switch Port/VLAN^Confidence) || … >

fc00:aaaa:aaaa:1111:217:31ff:feb8:F21C - - [28/Nov/2019:12:11:57 +0530] "GET / HTTP/1.1" 200 4335 "-" "Mozilla/5.0 (X11; U; Linux x64; en-US; rv:1.9.2.13) Gecko/20101206 Ubuntu/20.04 Firefox"

<[FC00AAAAAAAA1111021731FFFEB8F21C -> 001731B8F21C** (10.100.111.21/0007/0001^^^)]>**

fc00:aaaa:aaaa:1111:217:31ff:feb8:F21C - - [28/Nov/2019:12:11:57 +0530] "GET /style.css HTTP/1.1" 200 1875 "http://[fc00:aaaa:aaaa:2222:20c:29ff:fe7a:32D5]/" "Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.9.2.13) Gecko/20101206 Ubuntu/20.04 Firefox"

<[FC00AAAAAAAA2222020C29FFFE7A32D5 -> 000C297A3(Port Unknown) || FC00AAAAAAAA1111021731FFFEB8F21C -> 001731B8F21C** (10.100.111.21/0007/0001^^^)]>2D5******

Figure 4.19: Output of Apache access log processed by ipac_textlog.py.

Bold portions represent text added by theIPAC processing script.

Nov 28 12:11:58 10.100.112.2 kernel: [6740.432560] [LANB_IN-2-R] IN=eth4 OUT=eth5 SRC=2001:0db8:1111:1111:020c:29ff:fe39:4C39 DST=2001:0db8:9999:9999:020c:29ff:feb7:723a LEN=104 TC=0 HOPLIMIT=63 FLOWLBL=0 PROTO=ICMPv6 TYPE=128 CODE=0 ID=4932 SEQ=2 **<[20010DB811111111020C29FFFE394C39 -> 000C29394C39**** (Port Unknown) || 20010DB899999999020C29FFFEB7723A -> Unknown]>**

Nov 28 12:11:59 10.100.112.2 kernel: [6741.470878] [LANB_IN-2-R] IN=eth4 OUT=eth5 SRC=2001:0db8:1111:1111:020c:29ff:fe39:4C39 DST=2001:0db8:9999:9999:020c:29ff:feb7:723a LEN=104 TC=0 HOPLIMIT=63 FLOWLBL=0 PROTO=ICMPv6 TYPE=128 CODE=0 ID=4932 SEQ=3 **<[20010DB811111111020C29FFFE394C39 -> 000C29394C39**** (Port Unknown) || 20010DB899999999020C29FFFEB7723A -> Unknown]>**

Figure 4.20: Output of iptables log processed by ipac_textlog.py . Bold portions represent text added by the IPAC processing script

The script ipac_pcaplog.py can process binary packet capture data (shown in Figure 4.22) andgenerate a separate IPv6 address text report for these packages (shown in Figure 4.21). Each IPv6 packet in the file has been identified and attempted to match the script with the MAC address and the physical switch port by each unicast source and destination address. If an IPv6has no MAC or switch port associated with this, the text has been replaced with "Unknown." The output of the script has been structured as follows:

Packet #

Source IPv6 Address

MAC Address (Switch ID/Switch Port/VLAN)

Destination IPv6 Address

MAC Address (Switch ID/Switch Port/VLAN)

...

```
Packet 1
FE80:0000:0000:0000:020C:29FF:FE39:4C39
    00:0C:29:39:2B:49 (Port Unknown)
2001:0470:C2DD:1010:4261:86FF:FEC7:3B4C
    40:61:86:C7:DA:24 (10.100.111.21/0005/0001)
Packet 2
2001:0470:C2DD:1010:4261:86FF:FEC7:3B4C
    40:61:86:C7:DA:24 (10.100.111.21/0005/0001)
FE80:0000:0000:0000:020C:29FF:FE39:4C39
    00:0C:29:39:2B:49 (Port Unknown)
```

```
Packet 8
FE80:0000:0000:0000:4261:86FF:FEC7:3B4C
    40:61:86:C7:DA:24 (10.100.111.21/0005/0001)
FE80:0000:0000:0000:020C:29FF:FE39:4C39
    00:0C:29:39:2B:49 (Port Unknown)
Packet 9
FE80:0000:0000:0000:020C:29FF:FE39:4C39
    00:0C:29:39:2B:49 (Port Unknown)
FE80:0000:0000:0000:4261:86FF:FEC7:3B4C
    40:61:86:C7:DA:24 (10.100.111.21/0005/0001)
```

```
Packet 13
2001:0470:C2DD:1010:4261:86FF:FEC7:3B4C
    40:61:86:C7:DA:24 (10.100.111.21/0005/0001)
2001:04          70:C2DD:1010:020C:29FF:FE39:4C39
    00:0C:29:39:2B:49 (Port Unknown)
Packet 14
2001:0470:C2DD:1010:020C:29FF:FE39:4C39
    00:0C:29:39:2B:49 (Port Unknown)
2001:0470:C2DD:1010:4261:86FF:FEC7:3B4C
    40:61:86:C7:DA:24 (10.100.111.21/0005/0001)
```

Figure 4.21: Report produced by ipac_pcaplog.py, generated from the packets

shown in Figure 4.22.

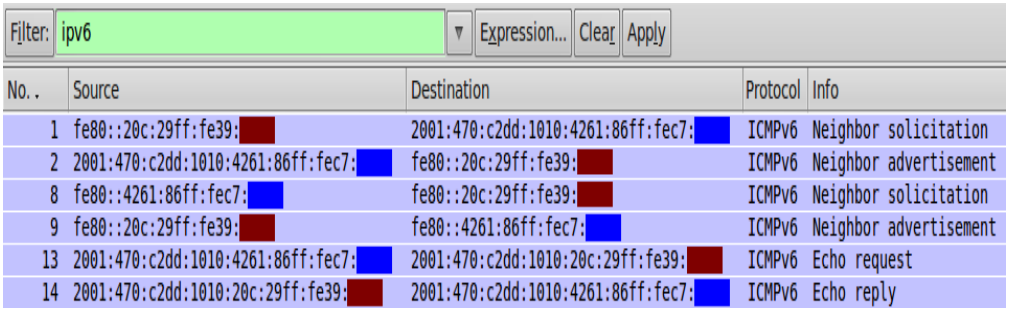


Figure 4.22: Packets captured on LAN A, viewed with Wireshark.

This IPv6 addressing information stored in the IPAC system was displayed in the iPac_graph.py script, as illustrated in Figure 4.23. Each known IPv6 address has been trackedthroughout the Y-axis and through time along the X-axis. The blue horizontal bar is placed on

the Y-axis beside the proper IPv6 label covering the correct starting and ending times of the x-axis in each IPv6-MAC pair in the IPAC Server records. Each bar has its corresponding MACaddress and has finished with a colored marker that shows the IPv6-MAC pair ending status: green when the IPv6/MAC session system is still active and red when it has ended. The graphshows the exact beginning and ending times of the session if a user clicked on a timeline bar (shown on the bottom bar in Figure 4.23). Any conflicts in the MAC have been marked by overlapping bars, as illustrated in Figure 4.23 in the fourth bar from the bottom.

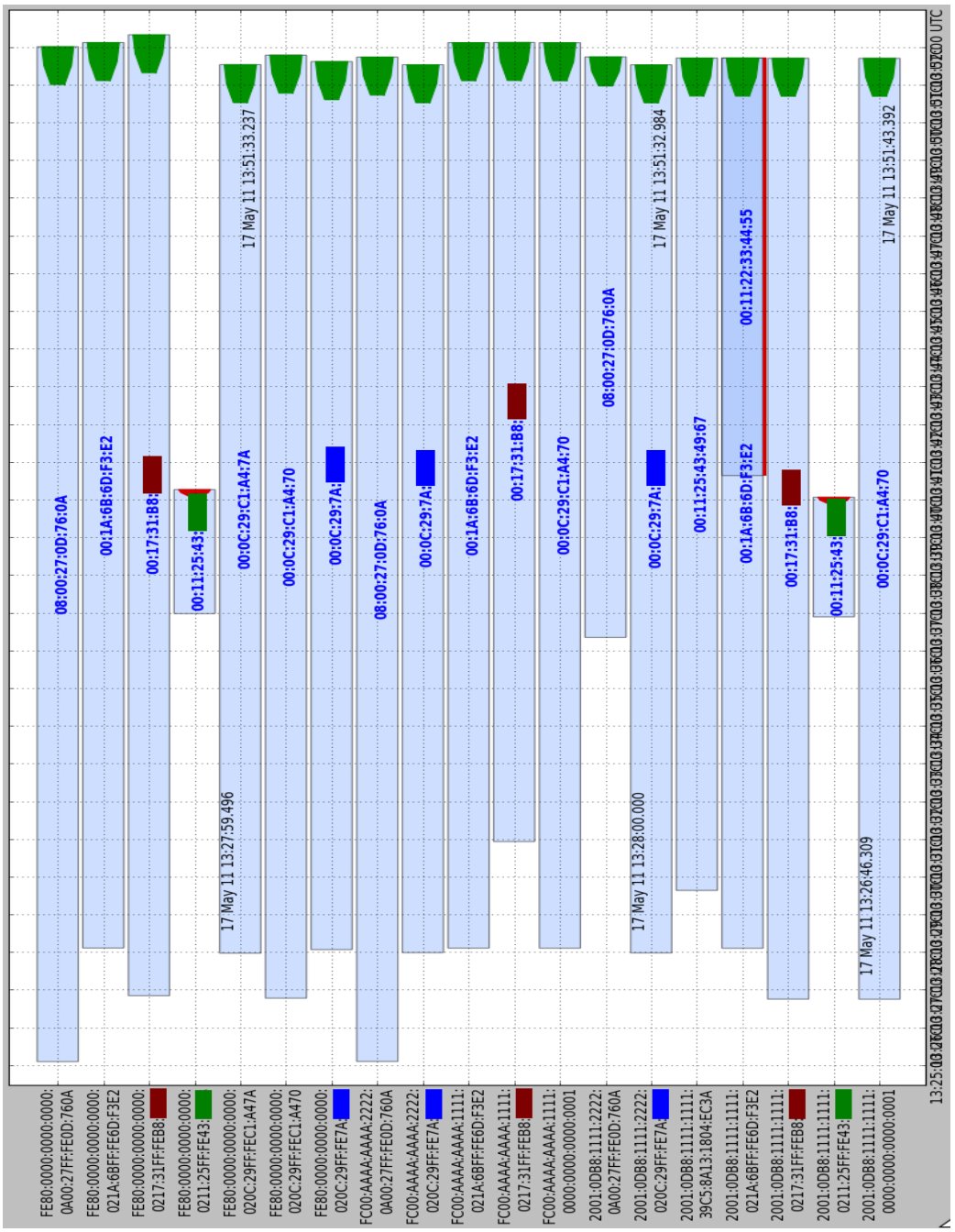


Figure 4.23: Interactive timeline graph of IPv6 and MAC address usage, produced by ipac_graph.py.

4.2 Implementation of DHCPv6

4.2.1 Tunnel/Tiny DHCPv6

DHCPv6 may be used to distribute information about IP addresses, prefix delegation information (aka routing), DNS server information, and a host of additional settings. The mainobjective of TDHCP is the exchange of information between server and client on the minimumnecessary for a Tunnel or PPP interface.

TDHCP is specific to work with a server on one side and a client on the other in tunnel-like settings – like a VPN or a PPP connection. Shared media, such as Ethernet, don't work well. Apart from all DHCPv6 implementations (like WIDE or ISC), its server only assumes to speakto a particular client and therefore always answers the same information, its client takes for granted that there can be only one server and that it fails with more than one. The client believesthat everything the server tells it is not valid, for example, it doesn't have duplicate address detection (DAD).

The following functions are implemented by TDHCP:

- Assignment address - tell the client what its IPv6 address is.
- Delegation prefix - tell the client which subnet to route via this link - tell the client aboutthe name servers.
- DNS Server and search domain.

The TDHCP client does not do any installation by itself - a script is called that should do the setup - assuming that the script has enough information about its environment to do the setup actually (eg. in a PPP environment it also receives the variables from PPP if tdhcpc is called from pppd).

Type tdhcpd —help or tdhcpc —help for the information on what TDHCP options areavailable.

4.2.2 DUID

TDHCP has its way of dealing with DUIDs as a very dynamic implementation. Based on the TDHCP starting time, link Layer-plus-time (LLT) addresses are not viable. Addresses based on Link-Layer (LL) also fail as TDHCP cannot guess what interface

ID are going to use and which interfaces are supposed to serve do not have LL IDs. This leaves DUIDs based on Enterprise number (EN).

TDHCP uses PEN 34360, adds several bytes to distinguish it from a TDHPT server or client (inner project ID: 0x0000) and a single byte of 0x01 or 0x00) and an MD5 sum of the user ID

of the user, to ensure that users can easily get their PEN from the IANA and develop a schemefor sub classifying it (or as fall-back the local hostname). If it is not suitable then it can be override the DUID with TDHCP.

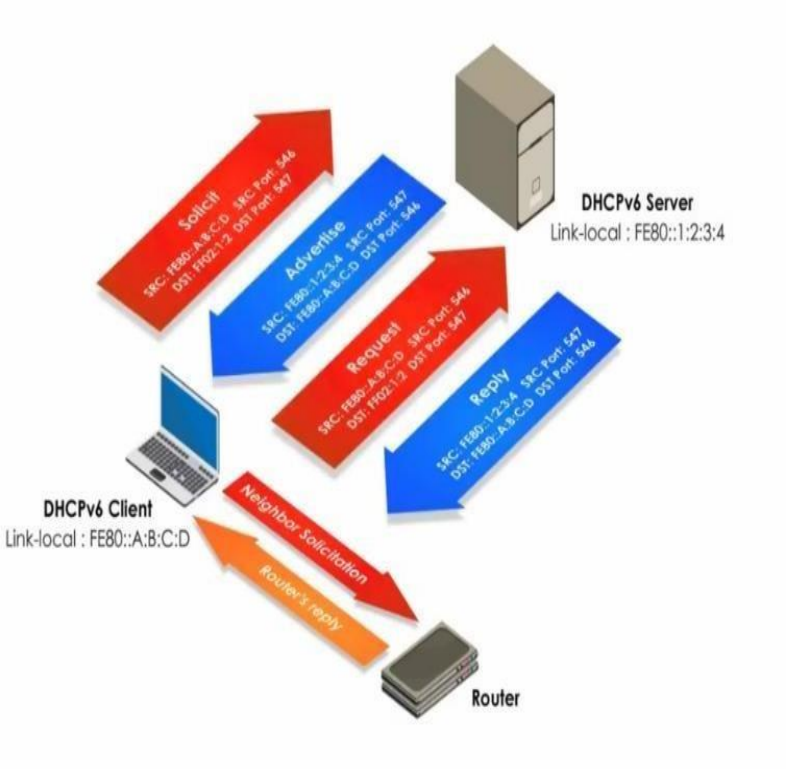


Figure 4.24: DHCPv6 Implementation

The proposed Python Code/Toolkit to implement the DHCPv6 using the Client MAC andother functions listed below because of the identified constraints:

Supported DHCPv6 Options

Clients and servers currently support the following DHCPv6 options. These options areoutlined in RFC 3315 if not stated otherwise. The purpose of this list is to increase.

1 – Client Identifier Option

2- Server Identifier Option

3 – Identity Association for Non-temporary Addresses Option

4 – Identity Association for Temporary Address Option

5 – IA Address Option

6 – Option Request Option

7 – Preference Option

12 – Server Unicast Option

13 – Status Code Option

14 – Rapid Commit Option

16 – Vendor Class Option

23 – DNS Recursive Name server Option

24 – Domain Search List

32 – Information Refresh Time

39 – Client FQDN Option

56 – NTP Server Optio

4.2.3 DHCPv6 Client Side:

DORA the DHCPv6 Client

A Python DHCP client command line designed to solve problems. Provides a port for thetransmission and inspecting of a response by tailored DHCP packets to a DHCP server.

Python Installation

Python3.8 directly, install with pip:

```
pip install dora-dhcp-client
```

Figure 4.25: install DHCP client

Alternate Installation

Install and run as a Docker image alternatively.

```
sudo curl -L --fail https://raw.githubusercontent.com/vfrazao-ns1/dora_dhcp_client/master/run.sh -o
/usr/local/bin/dora

sudo chmod +x /usr/local/bin/dora
```

Figure 4.26: Alternate DHCP Installation

Requirements:

- Python 3.8.0 or higher
- Docker (Optional)

NOTE: The Ubuntu 18.04 and Windows WSL have tested this. Other platforms may ormay not work.

Basic Usage

Check if Port 68 is currently bound to another program (and Port 67 if relay field is set) (for example with: sudo netstat -tulpn). If something is attached to those ports, they must be killedbefore starting again and prevented.

Run dora.py with a -h/--help flag to see all the available options:

```
$ sudo dora.py -h
usage: dora.py [-h] [-i INTERFACE] [-a MAC_ADDR] [-d] [-u] [-s SERVER] [-r RELAY] [-v] [-o OPTIONS] [-p PORT] [--
target_port TARGET_PORT] [-@ TARGET]

optional arguments:
  -h, --help            show this help message and exit
  -i INTERFACE, --interface INTERFACE
                        Interface to bind to and make DHCP requests
  -a MAC_ADDR, --mac_addr MAC_ADDR
                        MAC address to use (default random)
  -d, --debug           Print debug statements
  -u, --unicast         Send DHCP packets over unicast to specified server
  -s SERVER, --server SERVER
                        Server to send DHCP packets. Required for unicast and for relay use.
  -r RELAY, --relay RELAY
                        Address to set the giaddr field to
  -v, --verbose         Verbosity level (v: show ack packet, vv: show all packets, vvv: show debug)
  -o OPTIONS, --options OPTIONS
                        JSON body of options to include in requests
  -p PORT, --port PORT  Port to send packets from on client machine
  --target_port TARGET_PORT
                        Port to send to on target machine
  -@ TARGET             Given an IP address of a DHCP server, sends unicast requests
```

Figure 4.27: dora options

NOTE: in order that dora.py may operate properly, it must be able to bind to port 68 (and 67 in some cases). Sudo may be necessary for this. Any service (e.g. systemd-networkd) already bound to such ports may also need to be stopped.

Dora.py runs without providing any options, it only binds to an arbitrary interface and sends UDP packets broadcast.

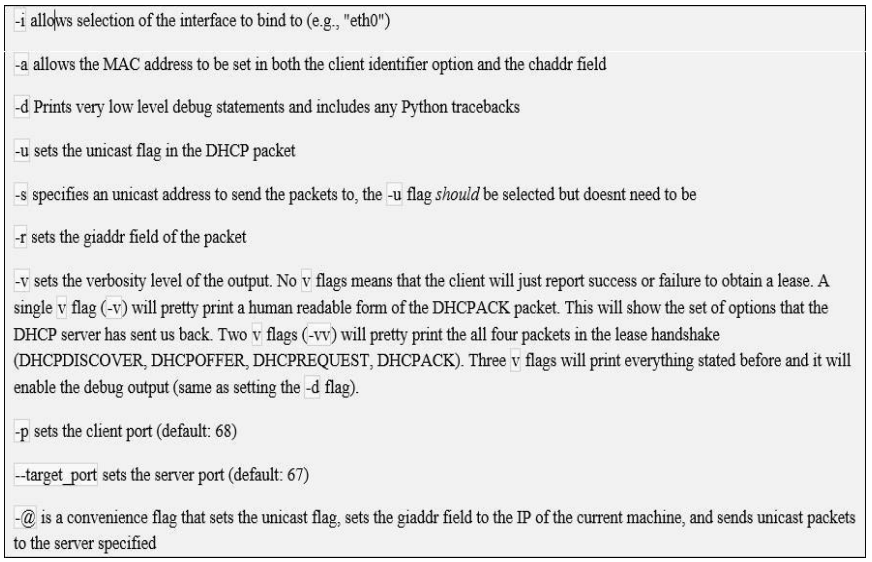


Figure 4.28: DHCP RFC 2131 sets the client port to 68 and the server port to 67

NOTE: the DHCP RFC 2131 sets the client port to 68 and the server port to 67 options that set different client or server ports are not expected to work with an RFC compliant server

MAC Binding in IPv6.

The main reason why dhcpy6d has been developed is because of the absence of RFC 3315 andall available implementations in the DHCPv6 server. This is an IPv6 display for dual-stack deployment in conjunction with DHCP infrastructure.

dhcpy6d solves this problem by tracking link-local addresses and related MAC addresses whenreading an application by reading the IPv6 neighbor cache (which works on local network segments, offline). Any request may be linked to a MAC address that can be used as a key forthe configuration information of a separate client in DHCP.

To get the information from the next cache, dhcpy6d tries, if the Link-Local Address and its MAC aren't known yet. The command /usr/sbin/ndp -a -n has been accessed on Linux since the
0.4 version. That's the way the next cache looks:

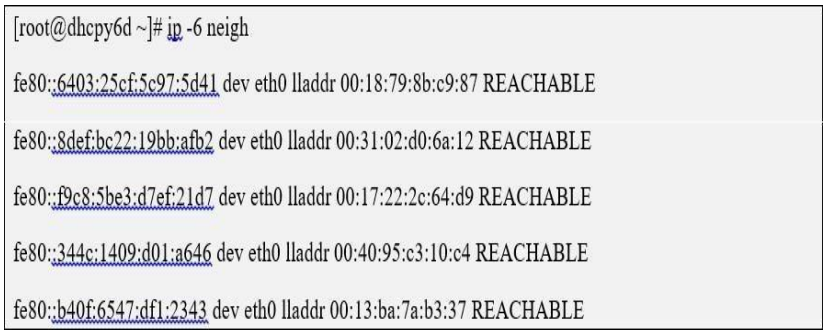


Figure 4.29: Cached MAC addresses

This occurs at the most once per unknown customer if the MAC is cached. As can be seen in the example above, in the next cache, several entries are all added to the internal mapping for dhcpy6d's Link-Local Address MAC. This means that the majority of customers are already familiar and are unable to trigger a new cache inspection. The overhead is therefore not too bigto call an external command.

If MACs' long caching is disabled due to cache poisoning security concerns, every new transaction scans the neighbor cache, which may lead to higher loading, but has not caused anyproblems in the tests.

If a cache or external call command does not yet show the client's Link-Local Address and MAC, the server sends out an answer that does not fit the one that is anticipated. "DHCPv6 StatusCode OK" is the message. The customer asks again and the cache next door has an opportunity for up-to-date data.

Dhcpy6d assigns it to the configured class when the client is identified. Each class has one ormore customer address schemes.

If the client is not known, a default address, error message, or no response will be provided.According to different patterns, addresses can be defined:

- Use a random address to generate a random address for use in internal networks.
- That can allow privacy on external connections even when the client does not haveaccess to privacy extensions.
- Build an internal ID in some kind of manageable database
- Use a MAC address to use in an internal network.

The general setup is set to the text file of a config.

Client settings can be stored in a configuration file or a database (MySQL or SQLite). In largerenvironments, databases will fit better.

In either MySQL or SQLite database, volatile information such as leases and Link-Local-Address-MAC-relations are stored.

Although MAC addresses are important, they can be omitted for certain dual-stack environments. For identification, DUIDs or hostnames may also be used.

Client and its DUID

Three ways to generate the DUID:

- Timestamp plus link layer,
- business ID plus variable portion, or
- Only the link-layer address (for Ethernet the MAC address).

The latter sounds like a MAC address, but not like IPv4's static assignment (only). The RFC reads: "This type of DUID comprises two bytes with a type 3 DUID, two bytes of network hardware code, and a link-layer address for a permanently connected network interface to the client or servers." The device should not include the MAC address of the card (alone and clearly) (Figure 1).

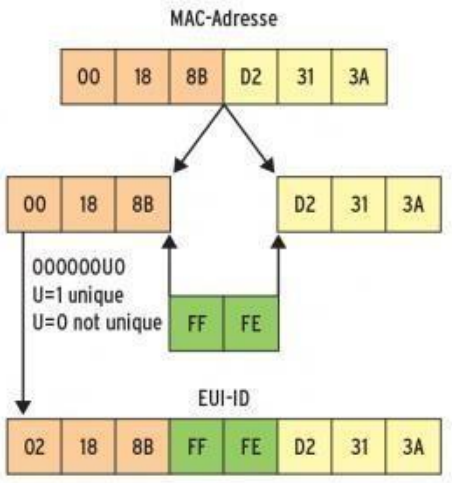


Figure 4.30: The IPv6 address ID of the MAC address has been created. FFFE is added and the second bit is toggledin the first byte.

The ICMPv6 protocol in Layer 3 determines the hardware address information in the world ofIPv6, which remains in Layer 2 with IPv4. This process was also called the Neighbor Discovery Protocol, by the developers (NDP, ("Neighbor Discovery Protocol", 2020)).

At first, the formal approach, without the difficult link to the MAC address, seems intelligent,as it makes a network device on all interfaces identifiable. However, this leads to problems in a virtualized world when a device is cloned. Because many implementations of DHCP clients only create the DUID value once and never compare it to the existing interfaces. On the positive, this means that after a network card is replaced, the manager must not change anythingin the DHCP configuration. When a virtual machine is cloned, the negative case is that all copies of the master get the same address as the DHCPv6 server.

4.3 OpenVPN with PKI Setup

The VPN server was configured in the first place by IPv4 and PKI and IPv6 were further integrated.

OpenVPN is a solution provided in Ubuntu repositories for Virtual Private Networking (VPN).It's soft, safe, and reliable. It is part of the SSL/TLS VPN stack family (different from IPSec VPNs). This topic covers the installation and setup of a VPN for OpenVPN.

To create a Public Key Infrastructure (PKI) to use SSL/TLS certification for authentication andkey exchange between the VPN server and clients, which is more than just pre-sharedOpenVPN keys. In routed or bridged VPN mode, OpenVPN can be operated with the

configuration of either UDP or TCP. The port number can be configured as well, but port 1194is the official port, for every communication this single port is used. VPN client implementations, including all Linux distributions, OS X, Windows, and OpenWRT-based wireless routers, are available for almost anything.

Server Installation

To install openvpn in a terminal enter:

```
sudo apt install openvpn easy-rsa
```

Figure 4.31: Install OpenVPN and easy-rsa

4.3.1 Public Key Infrastructure Setup

The first step to build an OpenVPN setup is to set up a PKI (public key infrastructure). ThePKI consists of:

- A separate certificate for the server and each client (a public key).

• Certificate and key Master Certification Authority (CA) used to subscribe to server andcustomer certificates.

- OpenVPN supports certificate-based bidirectional authentication, which means the client needsto authenticate the server certificate. Before establishing mutual trust, the server has toauthenticate the client certificate.
- The server and the client will authenticate the other by checking that the presented certificate has been signed first by the master certificate authority and then by checking in the now certified header information, like a common name or type of certificate (client or server).

4.3.2 Certificate Authority Setup

First copy an easy−rsa directory to /etc/openvpn to set up user defined Certificate Authority (CA) and generate OpenVPN server and multiple client certificates and keys. This ensures thatno changes are lost to the scripts after the package is updated. Run out of a terminal:

```
sudo make−cadir / etc /openvpn/easy−rsa
```

Figure 4.32: compile rsa certificate

Note: It is also possible to edit the file directly and adjust /etc/openvpn/easy−rsa/vars. The newly created /etc/openvpn/easy−rsa directory changes the root user, and runs:

```
./ easyrsa init−pki

./ easyrsa build−ca
```

Figure 4.33: Location of Certificate

Server Keys and Certificates

Next, the main pair for the server will be generated:

```
./ easyrsa gen−req myservername nopass
```

Figure 4.34: Generate Certificates -1

For the OpenVPN server, Diffie Hellman parameters must be generated. In pki/dh.pem thefollowing is placed.

```
./easyrsa gen-dh
```

Figure 4.35: Generate Certificates -2

Finally, a server certificate:

```
./easyrsa gen-req myservername nopass

./easyrsa sign-req server myservername
```

Figure 4.36: Server Certificate

In subdirectories, all certificates and keys were created. Copying them to /etc/openvpn/ iscommon practice:

```
cp pki/dh.pem pki/ca.crt pki/issued/myservername.crt pki/private/myservername.key /etc/openvpn/
```

Figure 4.37: Copy of Certificate

4.3.3 Client Certificates

A certificate to authenticate to a server is also required for the VPN client. Normally for each client, it is required to create a different certificate.

It can be done either on the server (as above keys and certificates) and then distributed safely to the client. Or vice versa: the client can generate, submit and sign a request from the server. Enter the following in a terminal while the user is a root to create the certificate:

```
./easyrsa gen-req myclient1 nopass

./easyrsa sign-req client myclient1
```

Figure 4.38: Client Certificate

Copies the .req file to the CA server if the first commands above were done on a remote system.Further pass the command like “easyrsa import−req /incoming/myclient1.req” for myclient1 toimport it. The second sign−eq command can then be followed.

In both cases, the following files are then copied to the client by a secure method:

- pki/ca.crt

- pki/issued/myclient1.crt

It is recommended to remove client certificates and keys from the server only on a clientmachine.

4.3.4 Simple Server Configuration

There is a sample config files (and a lot more details to check out) together with OpenVPNinstallation.

```
root@server :/# l s −l /usr/share/doc/openvpn/examples/sample−config−files/
total 68
−rw−r−−r−− 1 root root 3427 2019−11−04 18:09 c l i e n t . conf
−rw−r−−r−− 1 root root 4141 2019−11−04 18:09 s e r v e r . conf . gz
```

Figure 4.39: Server Configuration

Start the server.conf.gz copy and unpack to /etc/openvpn/server.conf.

```
sudo cp /usr/share/doc/openvpn/examples/sample−config −files/server.conf.gz /etc/openvpn/
myserver.conf.gz
sudo gzip −d /etc/openvpn/myserver.conf.gz
```

Figure 4.40: Copy Server Configuration file

To ensure that the following rows show the certificates and keys that was created in theprevious section, edit /etc/openvpn/myserver.conf.

```
ca ca.crt
cert myservername.crt
key myservername.key
dh dh2048.pem
```

Figure 4.41: Edit myserver.conf file

Finish this set with a “ta” key for tls-auth as in etc/openvpn:

```
sudo openvpn −−genkey −−secretta.key
```

Figure 4.42: tls-auth key

Edit the following line to allow IP forwarding by /etc/sysctl.conf and uncomment. Then reload sysctl

```
#net.ipv4.ip_forward=1
```

Figure 4.43: enable IPv4

```
sudo sysctl −p /etc/sysctl.conf
```

Figure 4.44: reload sysctl

Must configure this minimum to obtain an OpenVPN server that is working. The sample server.conf file can use all of the default settings. Run the server now.

Keep in mind that openvpn "systemctl start openvpn" is not starting. Openvpn uses system jobs templatized, openvpn@CONFIGFILENAME. For example, if myserver.conf is the configuration file, then the service is called openvpn@myserver. Use a templated service like openvpn@server for any type of service and systemctl commands such as start/stop/enable/disable/preset.

```
$ sudo systemctl start openvpn@myserver
```

Figure 4.45: Start service

Logging and error messages are available in the journal. For instance, filter for this particular message source using: if started a template service openvpn@server:

```
sudo journalctl −u openvpn@myserver −xe
```

Figure 4.46: Start template service

For all systemctl, the same templated approach works:

```
$ sudo systemctl status openvpn@myserver
openvpn@myserver . s e r v i c e − OpenVPN connection to myserver
Loaded : loaded (/ l i b /systemd/system/openvpn@ . s e r v i c e ; disabled ; vendor
p r e s e t : enabled )
Active : a c t i v e ( running ) s i n c e Thu 2019−11−04 19:49:45 UTC; 10 s ago
Docs: man: openvpn (8)
https :// community . openvpn . net /openvpn/ wiki /Openvpn24ManPage
https :// community . openvpn . net /openvpn/ wiki /HOWTO
Main PID : 4138 ( openvpn )
Status : " I n i t i a l i z a t i o n Sequence Completed"
Tasks : 1 ( l i m i t : 533)
Memory : 1.0M
CGroup : /system . s l i c e /system−openvpn . s l i c e /openvpn@myserver . s e r v i c e
4138 / usr / sbin /openvpn −−daemon ovpn−myserver −−s ta t us /run/openvpn
/ myserver . s ta t u s 10 −−cd / etc /openvpn −−s c r i p t −s e c u r i t y 2 –
c o n f i g / etc /openvpn/ myserver . conf −−writepid /run/
Nov 14 13:29:56 eoan−vpn−s e r v e r ovpn−myserver [ 4 1 3 8 ] : / sbin / ip addr add dev
tun0 l o c a l 1 0 . 8 . 0 . 1 peer 1 0 . 8 . 0 . 2
Nov 14 13:29:56 eoan−vpn−s e r v e r ovpn−myserver [ 4 1 3 8 ] : / sbin / ip route add 1 0 . 8 . 0 . 0 / 2 4 via 1 0 . 8 . 0 . 2
Nov 14 13:29:56 eoan−vpn−s e r v e r ovpn−myserver [ 4 1 3 8 ] : Could not determine IPv4/ IPv6 protocol . Using AF_INET
Nov 14 13:29:56 eoan−vpn−s e r v e r ovpn−myserver [ 4 1 3 8 ] : Socket Buffers : R =[212992−>212992] S=[212992−>212992]
Nov 14 13:29:56 eoan−vpn−s e r v e r ovpn−myserver [ 4 1 3 8 ] : UDPv4 l i n k l o c a l ( bound ) : [AF_INET ] [ undef ] : 1 1 9 4
Nov 14 13:29:56 eoan−vpn−s e r v e r ovpn−myserver [ 4 1 3 8 ] : UDPv4 l i n k remote : [ AF_UNSPEC]
Nov 14 13:29:56 eoan−vpn−s e r v e r ovpn−myserver [ 4 1 3 8 ] : MULTI: multi_init called , r=256 v=256
Nov 14 13:29:56 eoan−vpn−s e r v e r ovpn−myserver [ 4 1 3 8 ] : IFCONFIG POOL: base =10.8.0.4 s i z e =62, ipv6=0 Oct 24 10:59:26
eoan−vpn−s e r v e r ovpn−myserver [ 4 1 3 8 ] : IFCONFIG POOL LIST
Nov 14 13:29:56 eoan−vpn−s e r v e r ovpn−myserver [ 4 1 3 8 ] : I n i t i a l i z a t i o n Sequence Completed
```

Figure 4.47: Status of the service

On one system, to enable/disable different openvpn services are available, but can also let Ubuntu do this. The AUTOSTARTin /etc/default/openvpn configuration is available. The values are permitted as 'all,' 'none' or space-separated list of VPN names. "all" is supposed if empty. The name of the VPN is the name of the VPN setup file. i.e. /etc/openvpn/home.conf would be home. If systemd, this variable needs the systemctl daemon to be loaded and the openvpn Service restarted (if removed entries then may have to stop those manually).

After "systemctl daemon-reload" all dependent services that the generator created for conf filesin the conf /lib/systemd/system-generators/openvpn-generator restart the "generic" Openvpn. Check now if the OpenVPN interface tun0 has been created:

```
root@server : / etc /openvpn# ip addr show dev tun0
5 : tun0 : <POINTTOPOINT, MULTICAST, NOARP,UP, LOWER_UP> mtu 1500 qdisc fq_codel
s t a t e UNKNOWN group d e f a u l t qlen 100
l i n k /none
i n e t 1 0 . 8 . 0 . 1 peer 1 0 . 8 . 0 . 2 / 3 2 scope g l o b a l tun0
v a l i d _ l f t f o r e v e r p r e f e r r e d _ l f t f o r e v e r
inet6 fe80 : : b5ac : 7 8 2 9 : f31e :32 c5 /64 scope l i n k stable -privacy
v a l i d _ l f t f o r e v e r p r e f e r r e d _ l f t f o r e v e r
```

Figure 4.48: Check the IP of tun0

4.3.5 Simple Client Configuration

Various OpenVPN client implementations are available both with and without GUIs. Now usean OpenVPN client for Ubuntu-based command/service that is part of the same package as theserver. It is recommended to reinstall openvpn on the client machine:

```
sudo apt i n s t a l l openvpn
```

Figure 4.49: Client Installation

This time Copy the sample config file client.conf to /etc/openvpn/:

```
sudo cp / usr / share /doc/openvpn/ examples /sample-config -f i l e s / c l i e n t . conf / etc / openvpn/
```

Figure 4.50: Copy client file

To verify that the following lines point to those files, copy the following client keys and certificate files that was created in the section above, for example, “/etc/openvpn/” and edit “/etc/openvpn/client.conf”. Skip the path if the files found in /etc/openvpn/.

```
ca ca . c r t
c e r t myclient1 . c r t
key myclient1 . key
t l s −auth ta . key 1
```

Figure 4.51: Place client certificate files

And it is required to indicate the name or address of the OpenVPN server. Ensure the configuration of the keyword client. This makes customer mode possible.

```
c l i e n t remote vpnserver . example . com 1194
```

Figure 4.52: Remote connectivity

Start now with the same templated mechanism the OpenVPN client:

```
$ sudo systemctl s t a r t openvpn@client
```

Figure 4.53: Start service

As its done on the server lets check the status:

```
$ sudo systemctl s t at u s openvpn@client
openvpn@client . s e r v i c e − OpenVPN connection to c l i e n t
Loaded : loaded (/ l i b /systemd/system/openvpn@ . s e r v i c e ; disabled ; vendor p r e s e t : enabled )
Active : a c t i v e ( running ) s i n c e Thu 2019−10−14 17:12:25 UTC; 6 s ago
Docs : man: openvpn (8)
https :// community . openvpn . net /openvpn/ wiki /Openvpn24ManPage
https :// community . openvpn . net /openvpn/ wiki /HOWTO
Main PID : 1522 ( openvpn )
Status : ” I n i t i a l i z a t i o n Sequence Completed”
Tasks : 1 ( l i m i t : 533)
Memory : 1.3M
CGroup : /system . s l i c e /system−openvpn . s l i c e / openvpn@client . s e r v i c e
3616 / usr / sbin /openvpn −−daemon ovpn−c l i e n t −−st at u s /run/openvpn/
```

```
c l i e n t . st a tu s 10 −−cd / etc /openvpn −−s c r i p t −s e c u r i t y 2 −−c o n f i g
/ etc /openvpn/ c l i e n t . conf −−writepid /run/openvp
Nov 14 16:12:26 eoan−vpn−c l i e n t ovpn−c l i e n t [ 3 6 1 6 ] : Outgoing Data Channel :
Cipher ’AES−256−GCM’ i n i t i a l i z e d with 256 b i t key
Nov 14 16:12:26 eoan−vpn−c l i e n t ovpn−c l i e n t [ 3 6 1 6 ] : Incoming Data Channel : Cipher
’AES−256−GCM’ i n i t i a l i z e d with 256 b i t key
Nov 14 16:12:26 eoan−vpn−c l i e n t ovpn−c l i e n t [ 3 6 1 6 ] : ROUTE_GATEWAY 1 9 2 . 1 6 8 . 1 2
2 . 1 / 2 5 5 . 2 5 5 . 2 5 5 . 0 IFACE=ens3 HWADDR=52:54:00:3 c : 5 a :88
Nov 14 16:12:26 eoan−vpn−c l i e n t ovpn−c l i e n t [ 3 6 1 6 ] : TUN/TAP device tun0 opened
```

```
Nov 14 16:12:26 eoan-vpn-client ovpn-client[3616]: TUN/TAP TX queue length set to 100
Nov 14 16:12:26 eoan-vpn-client ovpn-client[3616]: /sbin/ip link set dev tun0 up mtu 1500

Nov 14 16:12:26 eoan-vpn-client ovpn-client[3616]: /sbin/ip addr add dev tun0 local 10.8.0.6 peer 10.8.0.5

Nov 14 16:12:26 eoan-vpn-client ovpn-client[3616]: /sbin/ip route add 10.8.0.1/32 via 10.8.0.5

Nov 14 16:12:26 eoan-vpn-client ovpn-client[3616]: WARNING: this configuration may cache passwords in memory -- use the auth-nocache option to prevent this

Nov 14 16:12:26 eoan-vpn-client ovpn-client[3616]: Initialization Sequence Completed
```

Figure 4.54: Check Service Status

The incoming connection appears on the server log. The name and address of the client as wellas success/failure messages can be seen.

```
ovpn-myserver[4818]: 192.168.122.114:55738 TLS: Initial packet from [AF_INET]192.168.122.114:55738, sid=5e943ab8 40ab9fed
ovpn-myserver[4818]: 192.168.122.114:55738 VERIFY OK: depth=1, CN=Easy-RSA CA
ovpn-myserver[4818]: 192.168.122.114:55738 VERIFY OK: depth=0, CN=myclient1
ovpn-myserver[4818]: 192.168.122.114:55738 peer info: IV_VER=2.4.7
ovpn-myserver[4818]: 192.168.122.114:55738 peer info: IV_PLAT=linux
ovpn-myserver[4818]: 192.168.122.114:55738 peer info: IV_PROTO=2
ovpn-myserver[4818]: 192.168.122.114:55738 peer info: IV_NCP=2
ovpn-myserver[4818]: 192.168.122.114:55738 peer info: IV_LZ4=1
ovpn-myserver[4818]: 192.168.122.114:55738 peer info: IV_LZ4v2=1
ovpn-myserver[4818]: 192.168.122.114:55738 peer info: IV_LZO=1
ovpn-myserver[4818]: 192.168.122.114:55738 peer info: IV_COMP_STUB=1
ovpn-myserver[4818]: 192.168.122.114:55738 peer info: IV_COMP_STUBv2=1
ovpn-myserver[4818]: 192.168.122.114:55738 peer info: IV_TCPNL=1
ovpn-myserver[4818]: 192.168.122.114:55738 Control Channel: TLSv1.3, cipher TLSv1.3 TLS_AES_256_GCM_SHA384, 2048 bit RSA
ovpn-myserver[4818]: 192.168.122.114:55738 [myclient1] Peer Connection Initiated with [AF_INET]192.168.122.114:55738
ovpn-myserver[4818]: myclient1/192.168.122.114:55738 MULTI_sva: pool returned IPv4=10.8.0.6, IPv6=(Not enabled)
```

```
ovpn-myserver[4818]: myclient1/192.168.122.114:55738 MULTI: Learn: 10.8.0.6 -> myclient1/192.168.122.114:55738
ovpn-myserver[4818]: myclient1/192.168.122.114:55738 MULTI: primary virtual IP for myclient1/192.168.122.114:55738: 10.8.0.6
ovpn-myserver[4818]: myclient1/192.168.122.114:55738 PUSH: Received control message: 'PUSH_REQUEST'
ovpn-myserver[4818]: myclient1/192.168.122.114:55738 SENT CONTROL [myclient1]: 'PUSH_REPLY,route 10.8.0.1,topology net30,ping 10,ping-restart 120,ifconfig 10.8.0.6 10.8.0.5,peer-id 0,cipher AES-256-GCM' (status=1)
ovpn-myserver[4818]: myclient1/192.168.122.114:55738 Data Channel: using negotiated cipher 'AES-256-GCM'
ovpn-myserver[4818]: myclient1/192.168.122.114:55738 Outgoing Data Channel: Cipher 'AES-256-GCM' initialized with 256 bit key
ovpn-myserver[4818]: myclient1/192.168.122.114:55738 Incoming Data Channel: Cipher 'AES-256-GCM' initialized with 256 bit key
```

Figure 4.55: Check tun0 interface created

And the customer can see if a tun0 interface has been created:

```
$ ip addr show dev tun0
4 : tun0 : mtu 1500 qdisc fq_codel s t a t e UNKNOWN group d e f a u l t qlen 100
l i n k /none
i n e t 1 0 . 8 . 0 . 6 peer 1 0 . 8 . 0 . 5 / 3 2 scope g l o b a l tun0
v a l i d _ l f t f o r e v e r p r e f e r r e d _ l f t f o r e v e r
inet6 fe80 : : 5 a94 : ae12 : 8 9 0 1 : 5 a75 /64 scope l i n k stable –privacy
v a l i d _ l f t f o r e v e r p r e f e r r e d _ l f t f o r e v e r
```

Figure 4.56: Show device details

See if the OpenVPN server is possible to ping:

```
root@client : / etc /openvpn# ping 10.8.0.1
PING 1 0 . 8 . 0 . 1 ( 1 0 . 8 . 0 . 1 ) 56(84) bytes of data .
64 bytes from 1 0 . 8 . 0 . 1 : icmp_req=1 t t l =64 time =0.920 ms
```

Figure 4.57: ping local IP

NOTE: In the client's network, OpenVPN uses the first IP address, and only the IP can be pinged. For example, the ::1 address will be used when it was configured. A /24 used for the client network mask. The P-t-P address in the ip addr above does not usually respond to requestsfor a ping.

The routes are as follows:

```
$ ip route
default via 192.168.122.1 dev ens3 proto dhcp src 192.168.122.114 metric 100
10.8.0.1 via 10.8.0.5 dev tun0
10.8.0.5 dev tun0 proto kernel scope link  src 10.8.0.6
192.168.122.0/24 dev ens3 proto kernel scope link src 192.168.122.114
192.168.122. 1 dev ens3 proto dhcp scope link src 192.168.122.114 metric 100
```

Figure 4.58: Check IP route

4.3.6 Providing IPv6 internet access to OpenVPN Clients

The OpenVPN server in this example has

- ipv4 address 192.168.122.1 (This is ipv4 address for example. In case the customizedIP might be used if required or it is also possible to use publicly accessible address)
- ipv6 for its purposes (the exact address does not matter)
- an additional subnet ipv6/48: 2001:db8:p1:/18.

For the use of the openvpn tunnel, the /64 subnet out or out of this subnet is assigned todb8:f00:bebe:/64.

OpenVPN server configuration

If ipv6 traffic is routed to the OpenVPN Server in 2001:db8:f00:/48, the openvpn serverconfiguration are as follows:

```
# general settings for openvpn server
local 192.168.122.1
proto udp
```

```
dev tun
# setting up an openvpn server with certificates is covered here:
https://community.openvpn.net/openvpn/wiki/GettingStartedwithOVPN
ca myca.crt
cert myserver.crt
key myserver.key
dh dh1024.pem
# use 10.8.0.0/24 for the openvpn tunnel ipv4
server 10.8.0.0 255.255.255.0
# adittional settings for ipv6:
server-ipv6 2001:db8:f00:bebe::/64
push "route-ipv6 ::/0"
push "route-metric 2000"
```

Figure 4.59: Check config file

There are special options for ipv6 in the last three lines. Let's take a closer look at them.

```
server-ipv6 2001:db8:f00:bebe::/64
```

Figure 4.60: Add IPv6 configuration

This is our subnet of /64 out of our bigger 2001:db8:f00::/48. It is used by OpenVPN for the assignment of endpoint tunnel addresses. For instance, the server takes the name 2001:db8:f00:bebe::1/64 and an address 2001:db8:f00:bebe::1006/64 is assigned to a client.

```
push "route-ipv6 ::/0"
```

Figure 4.61: Add more settings for IPv6 configuration

This option enables the server to direct the client to::/0 traffic via the vpn on ipv6 (that is, the whole Internet of ipv6).

```
push "route-metric 2000"
```

Figure 4.62: Add more settings for IPv6 configuration-2

Set the default route metric for any network which is routed via the VPN to 2000 in the final line (both ipv4 and ipv6). 2000 is of very large value and therefore, if the client has a better ipv6 connection available, the path via openvpn to ipv6 internet will not be used. It is an optional line.

Finally, it is required to ensure that ipv6 packets can be routed by the Linux kernel:

```
sysctl sys.net.ipv6.conf.all.forwarding=1
```

Run the server command as follows:

Figure 4.63: Add more settings for IPv6 configuration-3

Add a line to /etc/sysctl.conf to make this option permanent across reboots:

```
net.ipv6.conf.all.forwarding=1
```

Figure 4.64: Add more settings for IPv6 configuration-4

Client configuration

To ensure safety, first a little firewalling: enable loopback interface connections, allowconnections initiated by the client, and drop everything else. Those are only three lines:

```
ip6tables -P INPUT DROP

ip6tables -I lo -j ACCEPT

ip6tables -A INPUT -m state --state RELATED,ESTABLISHED -m comment --comment "accept traffic
for established connections" -j ACCEPT
```

Figure 4.65: Firewall Configuration

Now the client configuration needs to be done, and it's just this:

```
client
dev tun-ipv6
remote 192.168.122.1
ca myca.crt
cert myclient.crt
key myclient.key
```

Figure 4.66: Configure Certificate Keys

Only the name of the tuning device is here special. The 'tun-ipv6' is named here. If no name isspecified, the tunnel device openvpns tunN will have a tun0, tun1, and so on, for a host with several tunnels. By naming the tunnel explicitly 'tun-ipv6,' remember that the network device is referred to as a 'tun-ipv6.' This allows simple ipv6 ip6tables firewalling.

The client is assigned an ipv6 address through OpenVPN when the OpenVPN connection hasbeen established and the client is ipv6-enabled.

4.4 Packet Customization using SDN – Software-defined Network.

SDN and Network Functions Virtualization (NFV) is two of the most dramatic technological changes in networking. Their designs, deployments, operations, and future networking and computer systems will be significantly altered. They will also determine the success (or failure)of suppliers and operators in the next five to ten years.

As with successful network technology, the industry standards and open systems will play a strong role in the timely adoption of SDN and NFV solutions and their ultimate success. Opensource will play an even more important role in fulfilling the promises of standardized and open networking.

OpenFlow is a protocol that is used to communicate from and to the network of southbound SDN controllers. OpenFlow is the protocol used to inform network switch's topology of flowsto their flow tables and advice switches on how to manage traffic flows not included in the present flow tables. Initially, there was no definition for OpenFlow to handle IPv6

communications. The next versions of OpenFlow now have IPv6 capability and more vendorsdeploy products using the newer versions of OpenFlow.

The separation of the control and the data plane is one of the key concepts to understand SDN.A network typically consists of many routers and switches, each

with information exchanged for topological purposes. Each of these network devices has its custom control system for brain-like functions such as trajectory or MAC. There also is a separate data plane for packet transmission in every network device. The challenge is that every device has a network perspective, and the only way to see this is through a CLI connection and the release of commands or configurations. The same goes for other devices such as firewalls, load balancers,and not just routers and switches.

The brain of the operations is a control plane. The software usually operates and constructs the necessary transmission tables, such as the RIB or MAC tables. Typically, this table is sent as acopy to the forwarding plane and installed in hardware to enable high traffic transmission. Bothof these planes are usually used on each network device in a network, and usually a management plane for things like ssh, snmp, and so on.

OpenFlow is an open standard for a communication protocol that allows the control unit to shutdown and interact with the forwarding plane of multiple devices from a

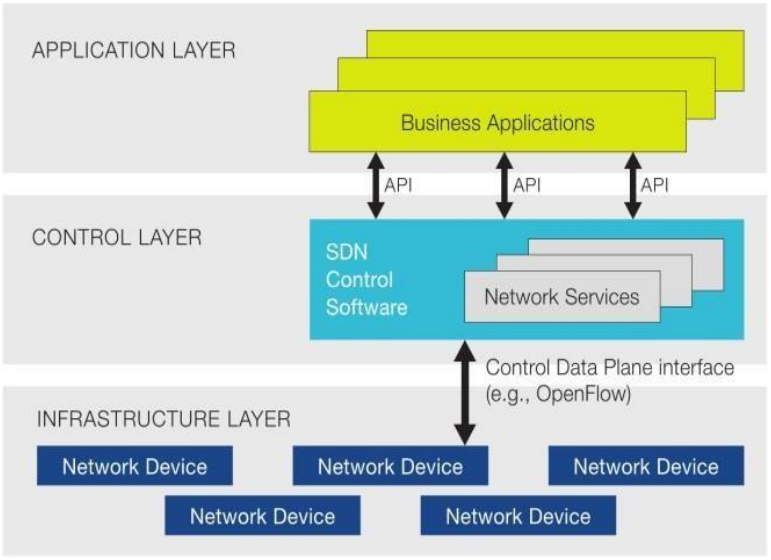


Figure 4.67: SDN Architecture

central point.

It should be noted before proceeding is that:

- SDN is not OpenFlow.
- SDN is much more than the data plane and the split control.

- SDN, like OpenFlow, is cool, but not the same.

4.4.1 A need for network abstraction

Application developers typically do not have to worry when writing applications about the hardware behind them. The operating system removed the hardware. Even the Operating

System itself has often been abstracted by hypervisors or containers from the hardware. This abstraction layer is a relatively new concept for the networking industry. OpenFlow createsan open interface for network abstraction layers as a freedom fighter.

The controller layer could be used to perform that abstraction. Without a connection to the network devices, which can manipulate flow tables and flow entries on network devices. An API is used by the application developer to communicate with the controller, and the controllerhandles the information required to update the flow tables of network devices. The applicationlayer is the beauty of SDN. OpenFlow is one way of (many) achieving the SDN abstraction.

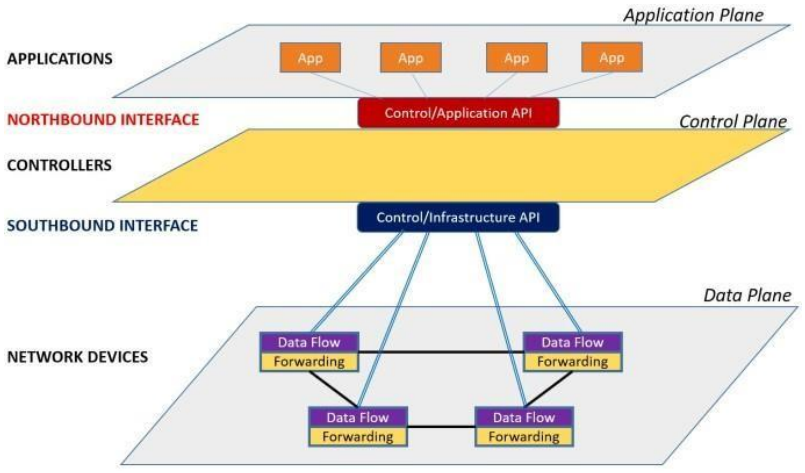


Figure 4.68: Software-Defined Networking - A high-level architecture

It should be noted here that OpenFlow does not update network device configurations. The flow tables of network devices are updated by OpenFlow. The OpenFlow logo makes it possible for packet A to reach packet B. It's no job for OpenFlow if required to configure NTPon network devices. Settings on devices that

have other procedures like SNMP, NETCONF, OVSDB, etc. will still need to be configured.

4.4.2 Components of an OpenFlow Switch

"One or more openflow logical switches are flow tables and group tables for packet searches and transmission, and one or more Openflow channels are provided to an external controller (Figure 1). The switch connects to the controller and manages the switch with the OpenFlow switch protocol."

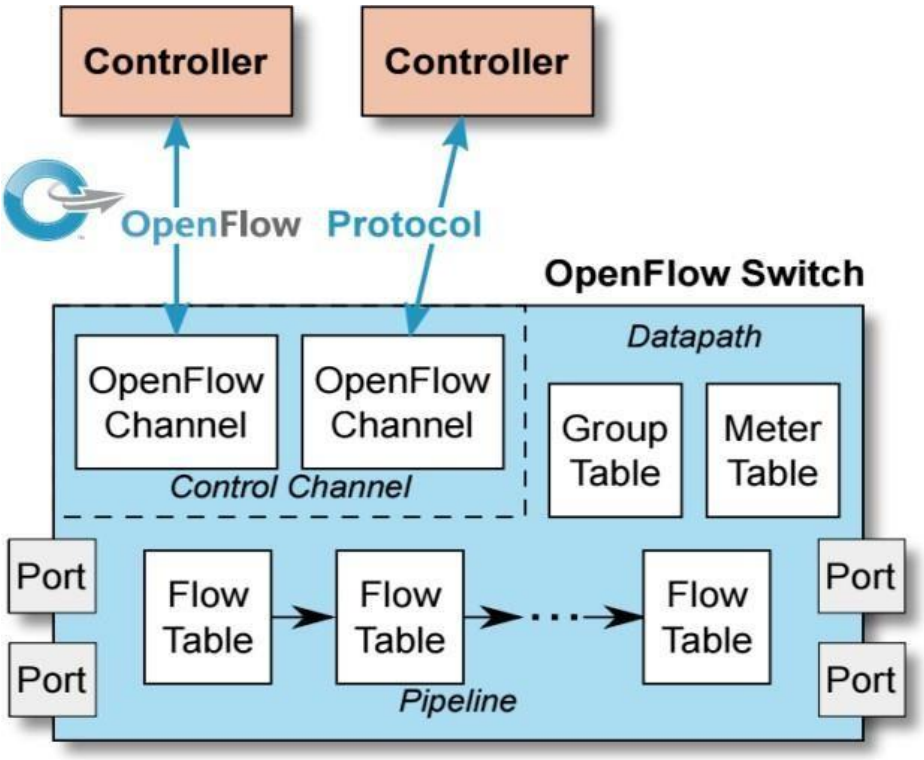


Figure 4.69: Main components of OpenFlow switch

4.4.3 Flow Tables

The controller can add, update and remove flow entries using the OpenFlow switch protocol, both in reactive (packet-responding) and proactive flow tables.

Responding Flow entries are created when the controller learns dynamically abouttopological devices and must update the flow tables on such devices to create end-to-end connectivity. Asswitches are simply a forwarding device of traffic in a pure OpenFlow environment, for example, the controller must first dictate and

program all the rational logic. There are also messages to the controller to find out how to get to the host if a host on switch A is to contact a host B switcher. The controller learns how and how to connect the host MAC address tablesand the logic in the flow tables for each switch. The controller flow entry is reactive.

Before traffic arrives, proactive flow entries are programmed. The controller can schedule theseflow entries on the OpenFlow endpoints in advance if it's known that both devices should or should not communicate.

4.4.4 Traffic Matching, Pipeline Processing, and Flow Table Navigation

Within the OpenFlow network, at least 1 flow table and several flow entries are included in each OpenFlow switch. These flow entries include match fields, counters, and packet instructions.

Typically have more than one flow table, so it's important to note that the match begins at the first flow table and can continue to the pipeline's additional flow tables. The packet will beginin Table 0 first and check the priority entries. First the highest priority (e.g. 200, then 100, then1). If the flow is to proceed to another table, go to update the packet details to the table in the instructions.

In two phases, Ingress processing, and Egress processing, this pipeline will take place.

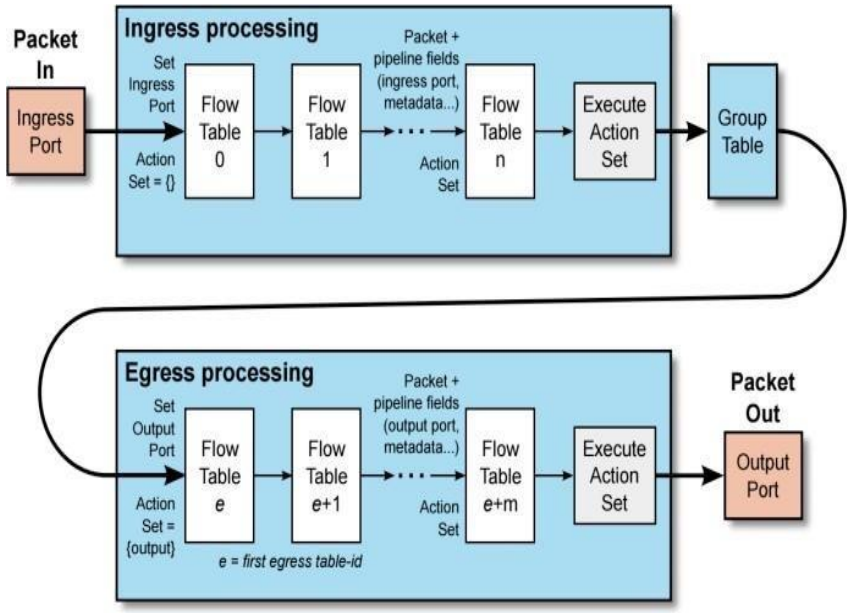


Figure 4.70: Two phases

The instructions for the specific flow entry are executed when a matching entry is found. If noflow table matches the results depend on the Table-miss flow entry configuration.

Table-miss flow entry

The last entry in the table is the table-miss flow, with a priority of 0 and a match for anything.The measures to be undertaken depending on how to configure them. This is a catch-all. Over the OpenFlow channel, can also forward the packet to the controller, remove it or continue to the next flow table.

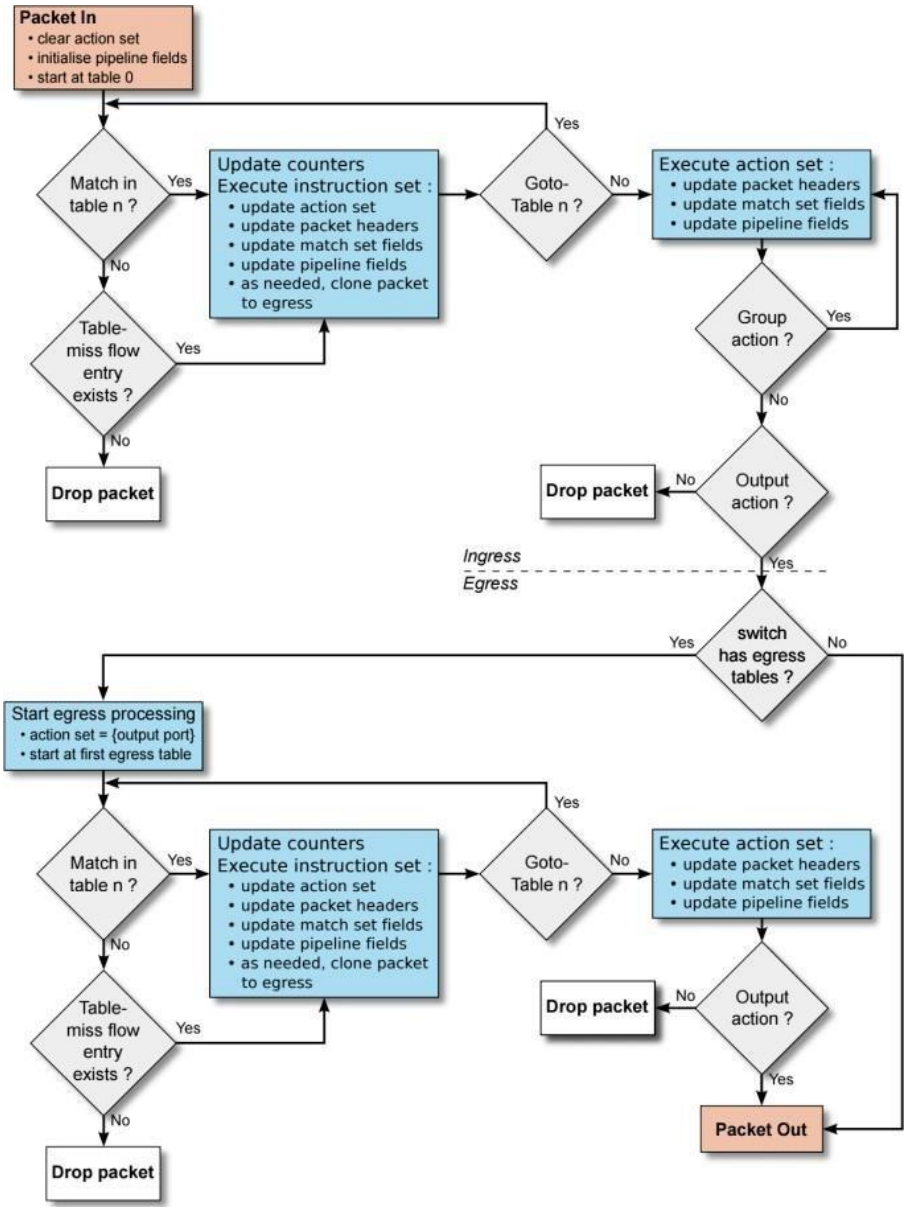


Figure 4.71:OpenFlow data Flow chart.

OpenFlow Ports

"OpenFlow ports are network interfaces between OpenFlow and the rest of the networks for passing packets. The logical connection of OpenFlow switches is with their OpenFlow ports..."The OpenFlow switch must support three types of ports: physical ports, logic ports, and reserved ports.

Physical Ports

Ports that correspond to a hardware interface on a switch are physical ports defined by a change.This might mean mapping OpenFlow's physical ports one-to-one on the switch to hardware- specific Ethernet interfaces but doesn't have to be individual. OpenFlow switches can have virtual physical ports and map to virtual physical port representation. This is similar to the wayhardware network interfaces are virtualized in computational environments.

Logical Ports

Logical ports are ports defined by the switch which do not match directly to the switch's hardware interfaces. For example, LAGs, tunnels, and loopback interfaces. The only differencebetween physical ports and logical ports is to have a packet associated with the logical port with an additional field of pipeline named Tunnel-ID and to inform the controller both of the logical port and the underlying physical port if packets are sent on logical ports that require communication to the controller.

Reserved Ports

The OpenFlow reserved ports provide generic transmission actions such as the transmission tothe controller, flood, or transmission using non-OpenFlow methods, such as "normal" processing.

ALL, CONTROLLER, TABLE, IN PORT, ANY, UNSET, LOCAL exist in various

parameters of the required reserved ports. The CONTROLLER Port is the OpenFlow channelused for switch/controller communication.
The NORMAL and FLOOD ports in hybrid environments permit interaction between OpenFlow and the switch's hardware pipeline.

OpenFlow-only Switches vs. OpenFlow-hybrid switches

Two OpenFlow switches are available: OpenFlow-only and OpenFlow-hybrid.

OpenFlow switches are "stupid switches" which have just a data/transmission device and no local decision-making option. The OpenFlow pipeline processes all packets, which cannot be otherwise processed.
OpenFlow hybrid switches support the operation of the OpenFlow, as well as the regular operation of the Ethernet switch. This means that the user can interact with the OpenFlow pipeline using various classification mechanisms using the traditional L2 Ethernet switching, VLAN isolation, L3 routing, ACLs, and QoS processing via the switch local control plane.
It is possible to use traditional routing and switching to have half of its ports, while the other half is configured for OpenFlow. The OpenFlow half would be controlled by an OpenFlow controller, the other half would be controlled by a local switcher. The use of a NORMAL orFLOOD port would be a necessary step in passing traffic between these pipelines.

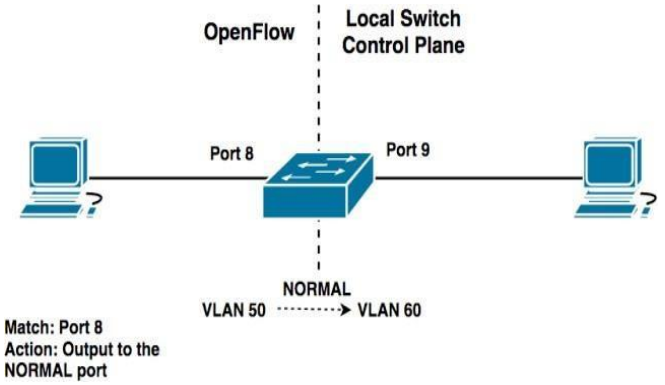


Figure 4.72: Data passing through channels

OpenFlow Messages

OpenFlow Protocol supports three types of messages, each with its subtype setup:

- Controller-to-switch

- Asynchronous
- Symmetric

Controller-to-switch Messages

Controller-to-switch Message initiated and used to manage or inspect the switch directly bythe controller. In the following messages:

- Functionality — switch to identity request
- Settings – Set and Query Settings
- Modify-Status – also known as 'flow mod' for adding, deleting, and modifyingflow/group entries
- Read-States – get statistical information
- Outs Package – controller sends the full packet or buffer ID message to the switch.
- Barrier – The controller uses request or reply messages to ensure that messagedependencies are fulfilled and that notifications are received.
- The role of OpenFlow channel – Role request
- Asynchronous Settings – set a filter to add an asynchronous message to OpenFlowChannel Settings

Asynchronous Messages

The switch will initiate asynchronous messages to update the network event controller andchange the switch state. The following messages include:

- Packet-in – transfer the control of a packet to the controller
- Flow-Removed – inform the controller that flow has been removed
- Port Status – inform the controller that the switch has gone down
- Error – notify the controller of problems

Symmetric Messages

Symmetric messages are initiated either by the switch or controller and sent without solicitation. The following messages include:

- Hello – Messages exchanged between switch and controller introduction ormaintenance
- Echo – These verify connection vitality and are used for measuring latency orbandwidth from either the switch or the controller
- Experimenter – A standard way to offer additional features within OpenFlow messagetype space for OpenFlow switches.

4.4.4.1.1 OpenFlow Connection Sequence

- A switch can initiate IP connectivity, a default transportation port, or a user-specifiedport (TCP 6633 pre-OpenFlow 1.3.2, TCP 6653 post).
- A controller can also initiate the connection request, but this isn't common.
- Established TCP or TLS connection
- Both send a populated version field with OFPT HELLO
- Both use evaluated the negotiated version.
- The message is sent in case it cannot be agreed upon an OFPT_ERROR
- The controller sends an OFPT_FEATURES-REQUEST to collect the Data path ID ofthe switch, along with the switch's capabilities, if both support the version.

Let's take a look at this process in the section below

Observing OpenFlow Messages in Wireshark

The process to capture OpenFlow traffic between mininet (OpenFlow Switch) and OpenDaylight (OpenFlow Controller).

- Mininet is at 192.168.0.11
- ODL is at 192.168.0.12

Initially execute the mininet, for the use of OpenFlow1.3, and further ping between host h1and host h2.

```
mininet@mininet-vm:~$ sudo mn --controller=remote,192.168.0.12 --mac --topo=single,2 --switch=ovsk,protocols=OpenFlow13
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1)
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet&gt; h1 ping h2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.746 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.365 ms
^C
--- 10.0.0.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.365/0.555/0.746/0.191 ms
```

Figure 4.73: ping between host h1 and host h2

On the OpenDaylight controller the topology produced in Mininet looks like this:

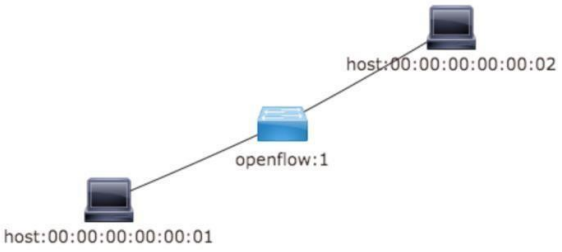


Figure 4.74: OpenFlow connection

Host 1 can be seen to be connected to s1-eth1 and host 2 to s1-eth2. Noticing that there is the LOCAL management interface used to send traffic directly to the control aircraft, the local management level of the switch.

Node Connector Id	Name	Port Number	Mac Address
openflow:1:2	s1-eth2	2	7A:78:1D:C5:7E:ED
openflow:1:1	s1-eth1	1	6A:32:D0:0C:30:09
openflow:1:LOCAL	s1	LOCAL	76:60:82:FA:A8:43

Figure 4.75: Connection Details

The topology of the mininet is available in the database Open vSwitch (OVS):

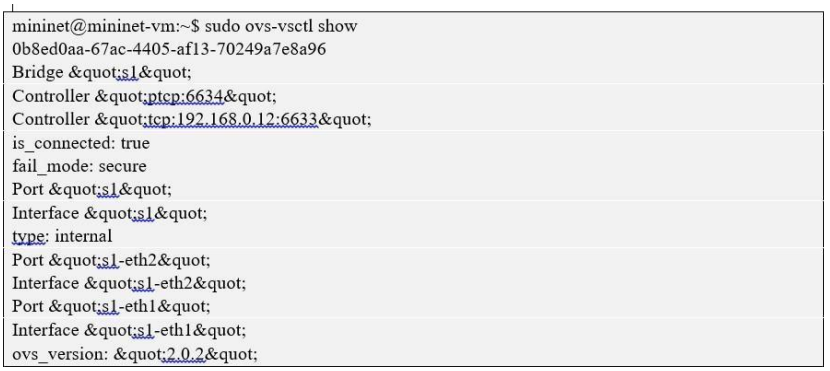


Figure 4.76: the database Open vSwitch (OVS)

The dumping flows in mininet can be observed:

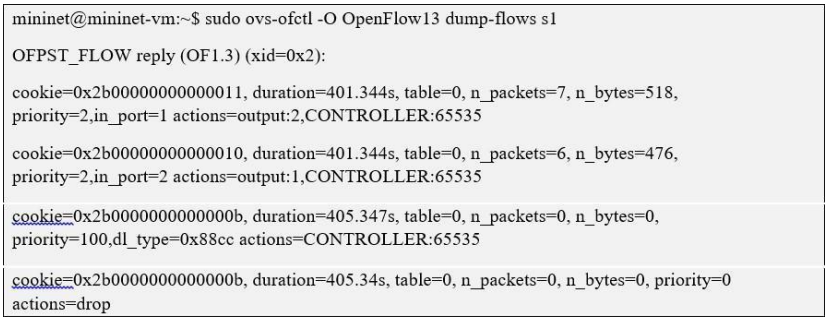


Figure 4.77: Mininet dumping flows

To view packets caught with an openflow v4 display filter using Wireshark (that meansOpenFlow 1.3).

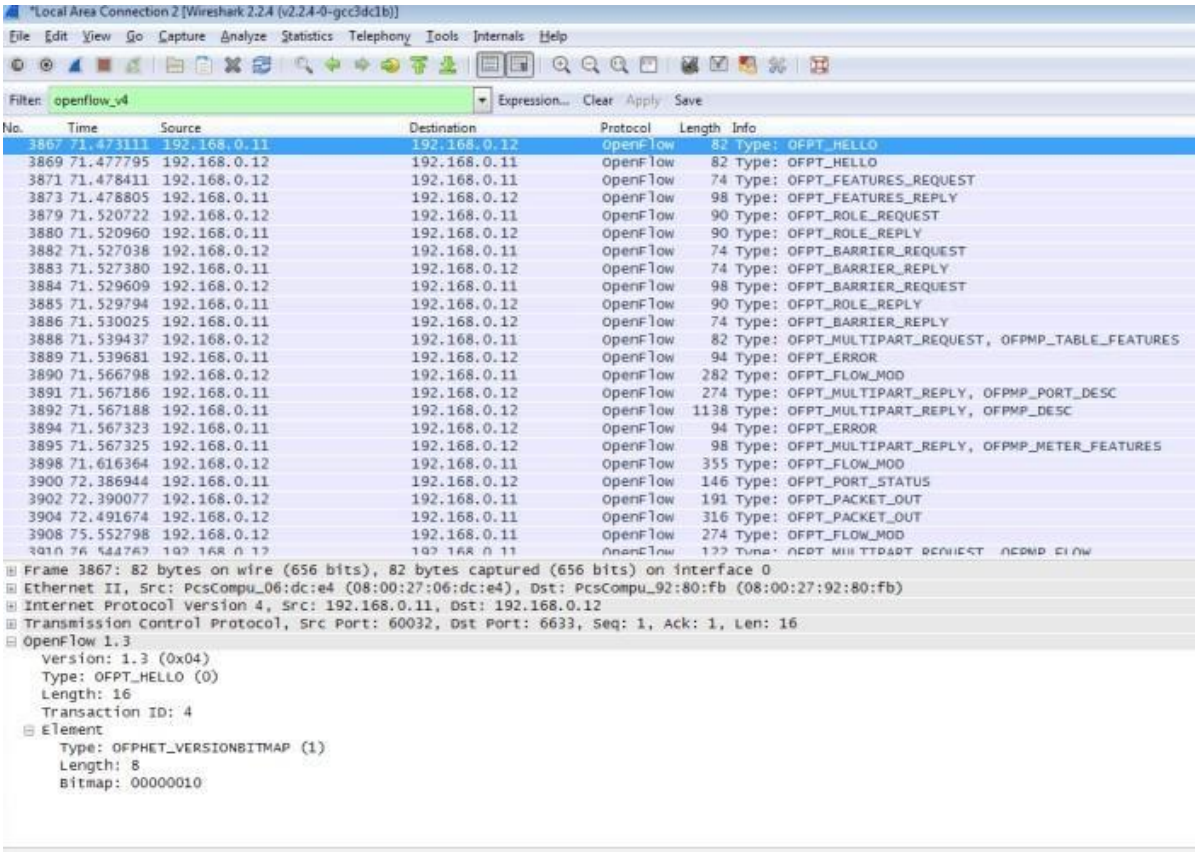


Figure 4.78: Dumping details in Wireshark

Note that a HELLO message from a switch to the controller (192.168.0.11) was the first packetto be caught (192.168.0.12). This is what the OpenFlow Specification document anticipates exactly. The OpenFlood Channel is initialized as the data path between the controller and the instrument in the OpenFlow protocol, which is the mininet switch in this case.

Again, the communication between the OpenFlow Channel switch and the controller identified.Note also that the control channel is used, which means that traffic uses the switch's LOCAL administration port and does not pass through a switch pipeline.

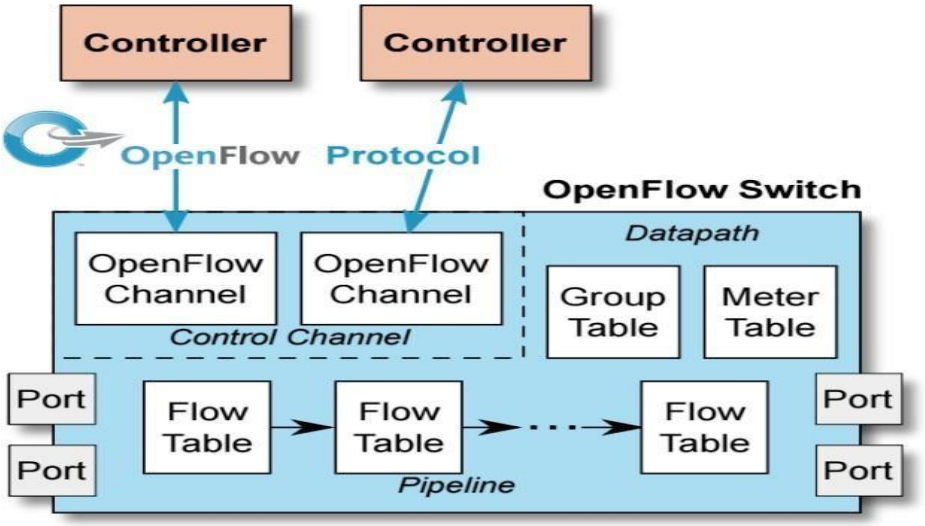


Figure 4.79: Main Components of an OpenFlow switch

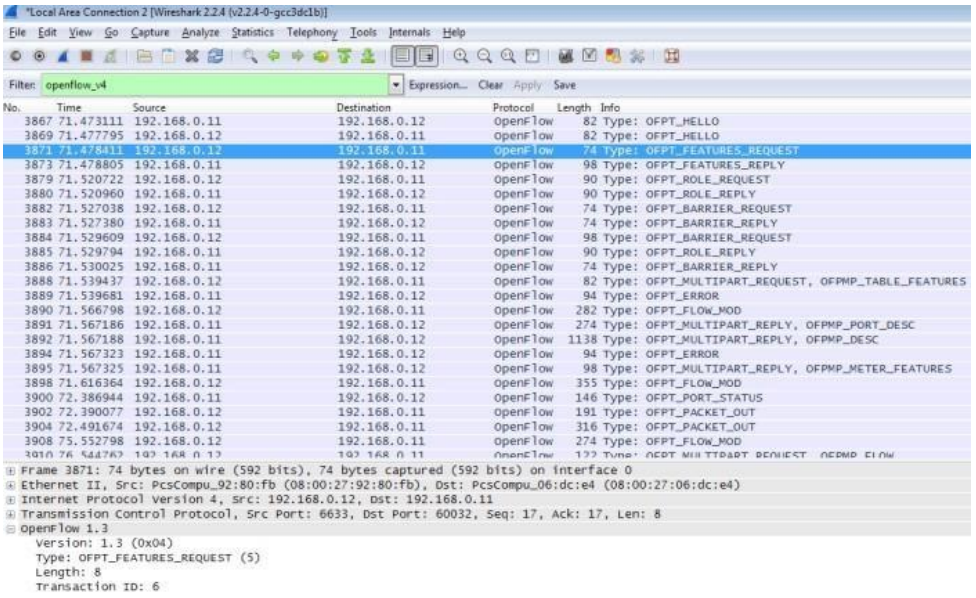


Figure 4.80: OFPT_FEATURES_REQUEST Packets

This message is used to find the data identifier (DPID) of the switch, among other capability attributes. This message. In OpenFlow Topology, the DPID identifies a data path uniquely andis dynamically developed, by combining the device MAC address in the lower 48 bits with a 16-bit string which the implementing device can determine.

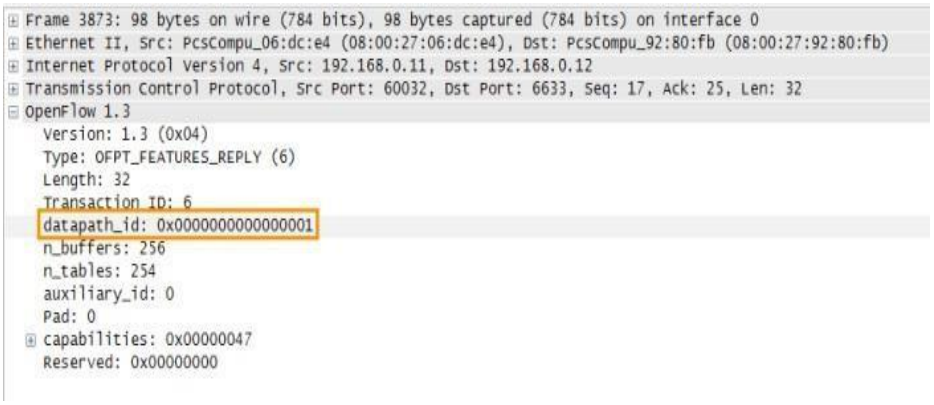


Figure 4.81: OFPT_FEATURES_REPLY

The switch replies with an OFPT_FEATURES_REPLY message.

n_tables

The OpenFlow specification specifies that the n_tables field describes the number of switch- supported tables, with different matching fields, actions, and entries that are supported. This isin the response message features as shown here:

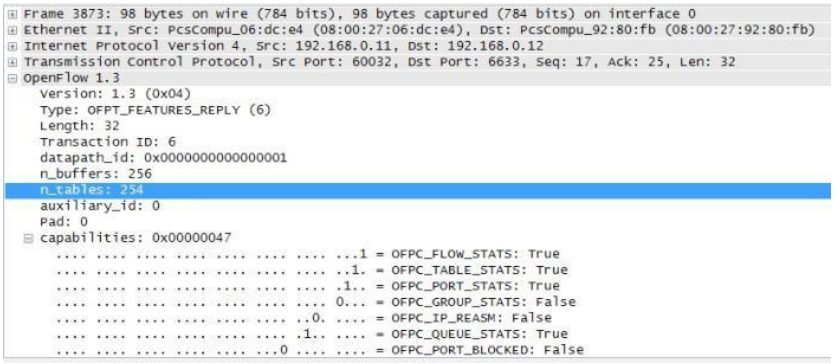


Figure 4.82: n_tables Information

If the controller has an idea of how large the tables are constructed, types, and orders, the controller sends a message OFFMP_TABLE_FEATURES. The switch then must return the tables as the packets pass through the tables.

Multipart Request

Multiple messages are used to encode requests or answers that can carry large quantities of data that may not fit in a single message. The regular messages are restricted to 64KB. The sequence is encoded for multi-part messages. These types of messages are used primarily to request information or statistics.

This exchange of information can be seen in our packet capture. Looking at the OFP_MULTIPART_REPLY package, let's see attributes like the producer and the versions ofhardware and software. OFPMP_DESC package

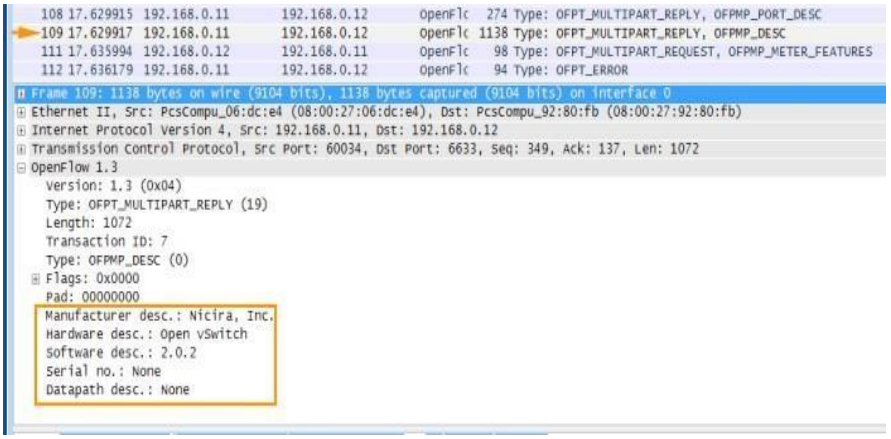


Figure 4.83: Frame Details

The message OFPT_PORT_DESC, OFPT_MULTIPART_REPLY lists information on the ports that the controller is sending. This message includes everything from type, status, and speed. The switch port remains down in that case.

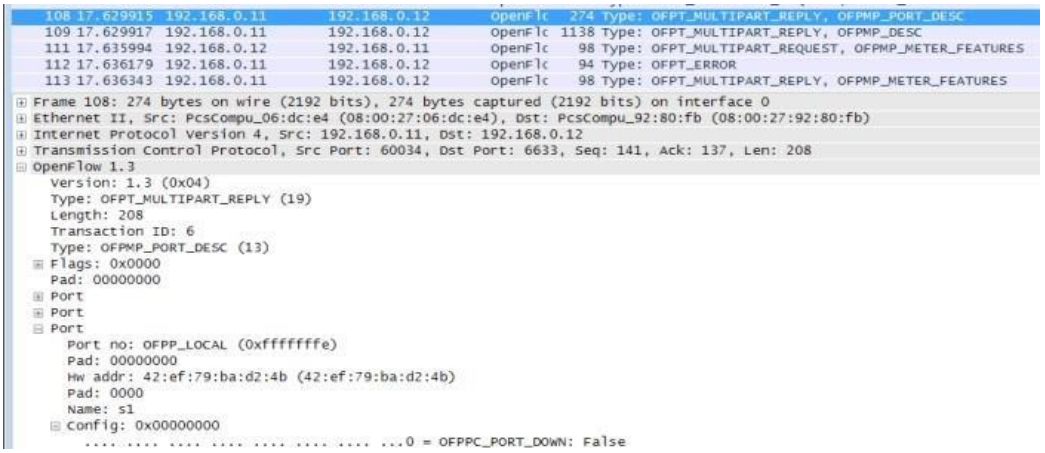


Figure 4.84: OFPT_PORT_DESC, OFPT_MULTIPART_REPLY

PACKET-IN and PACKET-OUT Messages

Messages from the PACKET_IN controller switch will be sent. I started a ping in the mininetfrom h1 to h2.

```
mininet&gt; h1 ping h2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.658 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.446 ms
^C
--- 10.0.0.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.446/0.552/0.658/0.106 ms
```

Figure 4.85: h1 ping h2

The Four PACKET_IN messages were found.

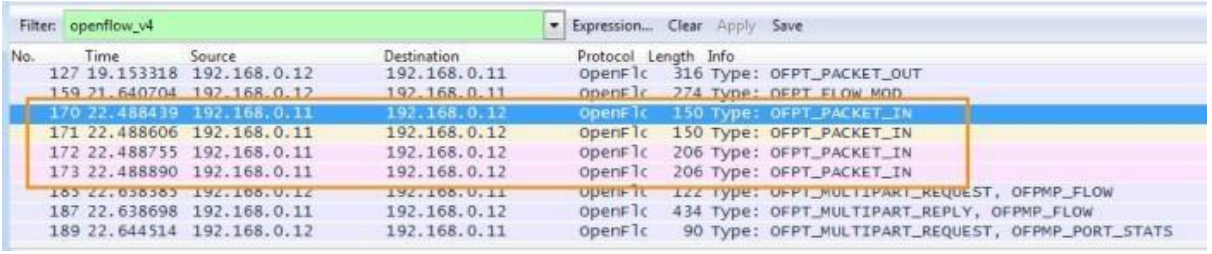


Figure 4.86: Four PACKET_IN messages

Found a PACKET_IN message from the controller switch first, because h1 still doesn't knowhow to get to h2. This is an ARP message that OpenFlow encapsulates via TCP.

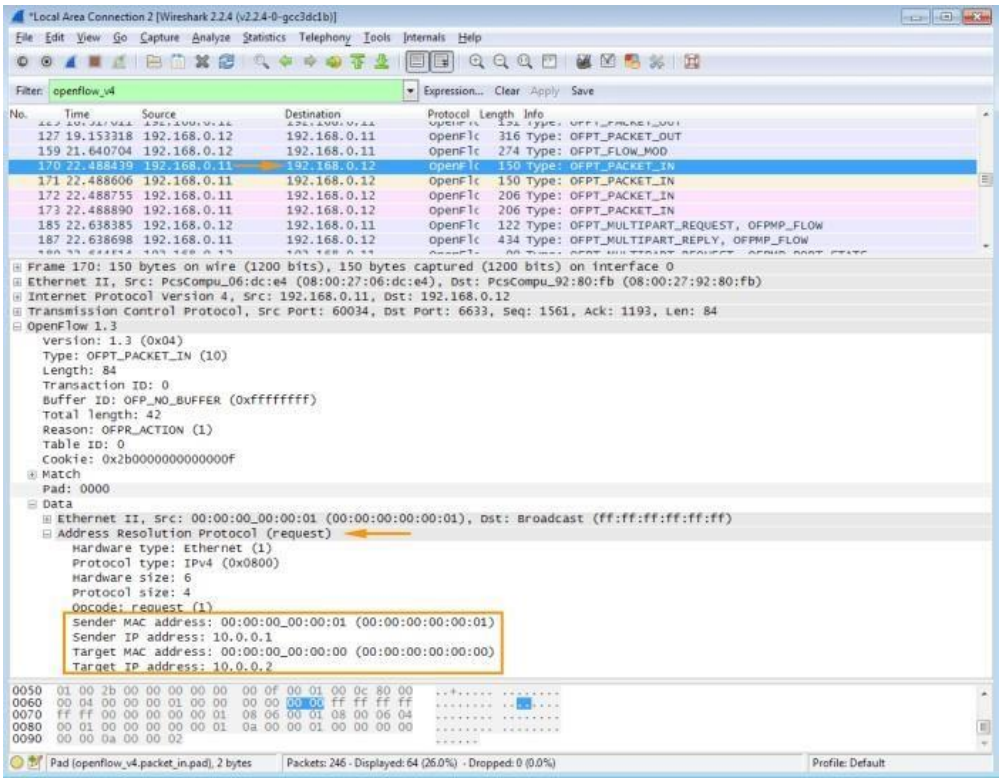


Figure 4.87: ARP Details

The ARP response from h2 is the next packet in consecution:

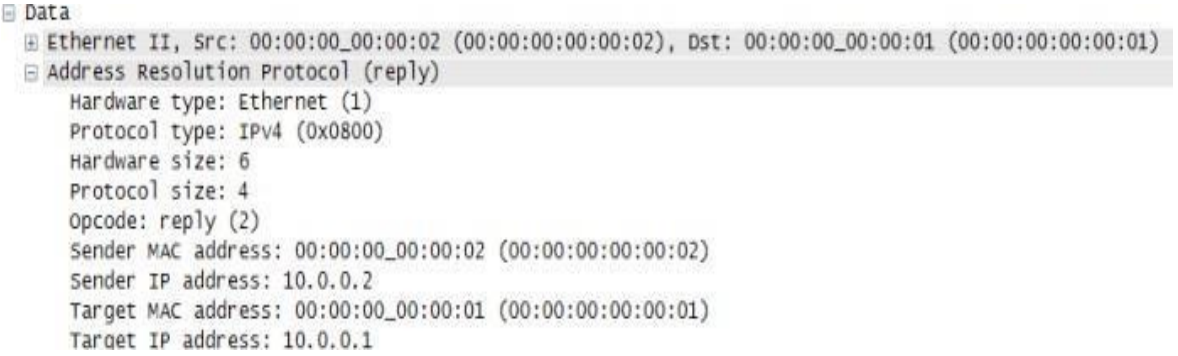


Figure 4.88: ARP response from h2

The ICMP echo is followed.

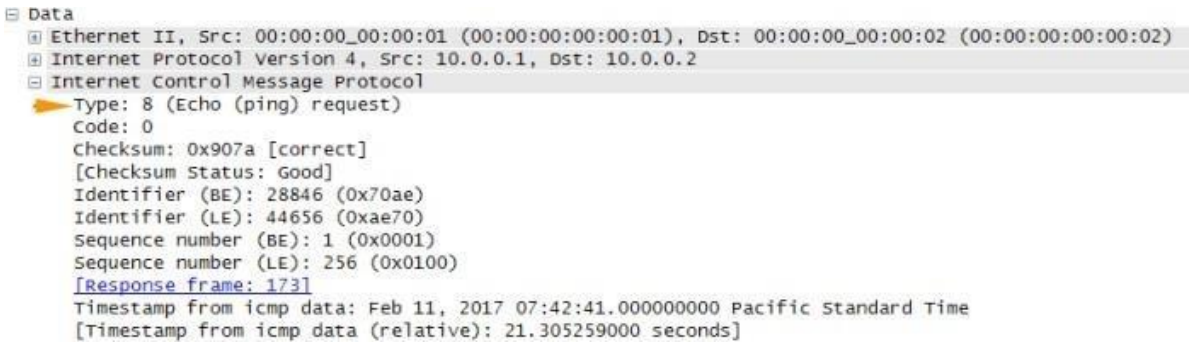


Figure 4.89: ICMP echo Packet

And there is an echo reply

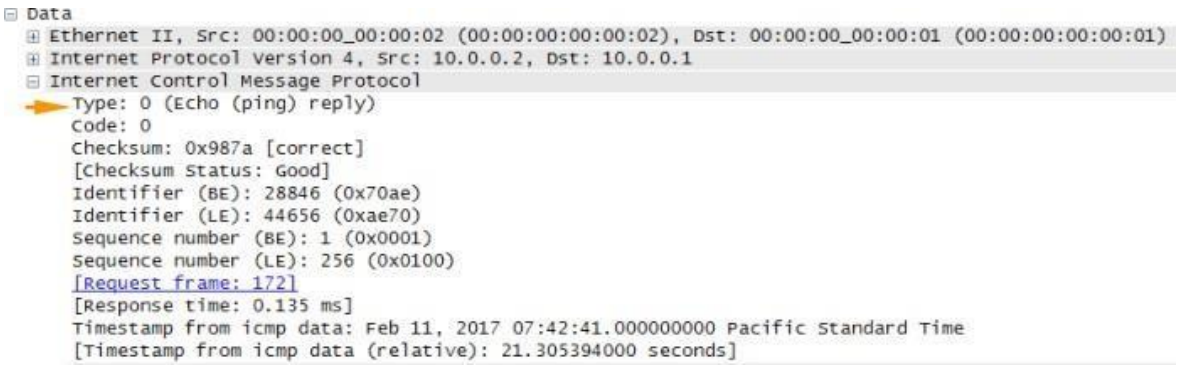


Figure 4.90: ICMP echo reply

The Open vSwitch (OVS) database shows these flows.

```
mininet@mininet-vm:~$ sudo ovs-ofctl -O OpenFlow13 dump-flows s1
OFPST_FLOW reply (OF1.3) (xid=0x2):
cookie=0x2a00000000000023, duration=394.662s, table=0, n_packets=18, n_bytes=2321,
idle_timeout=300, hard_timeout=600, priority=10,dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:02
actions=output:2
cookie=0x2a00000000000022, duration=394.662s, table=0, n_packets=19, n_bytes=1424,
idle_timeout=300, hard_timeout=600, priority=10,dl_src=00:00:00:00:00:02,dl_dst=00:00:00:00:00:01
actions=output:1
cookie=0x2b00000000000011, duration=401.344s, table=0, n_packets=7, n_bytes=518,
priority=2,in_port=1 actions=output:2,CONTROLLER:65535
cookie=0x2b00000000000010, duration=401.344s, table=0, n_packets=6, n_bytes=476,
priority=2,in_port=2 actions=output:1,CONTROLLER:65535
cookie=0x2b0000000000000b, duration=405.347s, table=0, n_packets=0, n_bytes=0,
priority=100,dl_type=0x88cc actions=CONTROLLER:65535
cookie=0x2b0000000000000b, duration=405.34s, table=0, n_packets=0, n_bytes=0, priority=0
actions=drop
```

Figure 4.91: Open vSwitch (OVS) database

If the flow is not hit, the idle timeout expires. Then the flow is deleted. It would not capture thepackets in the controller if it would fire up another relationship between these hosts as the flowsare already scheduled on the switch.

This applies if it is one topology switch or 50 topology switches.

```
mininet@mininet-vm:~$ sudo mn --controller=remote,192.168.99.104 --mac --topo=linear,50
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2 h3 h4 h5 h6 h7 h8 h9 h10 h11 h12 h13 h14 h15 h16 h17 h18 h19 h20 h21 h22 h23 h24 h25 h26 h27
h28 h29 h30 h31 h32 h33 h34 h35 h36 h37 h38 h39 h40 h41 h42 h43 h44 h45 h46 h47 h48 h49 h50
*** Adding switches:
s1 s2 s3 s4 s5 s6 s7 s8 s9 s10 s11 s12 s13 s14 s15 s16 s17 s18 s19 s20 s21 s22 s23 s24 s25 s26 s27 s28
s29 s30 s31 s32 s33 s34 s35 s36 s37 s38 s39 s40 s41 s42 s43 s44 s45 s46 s47 s48 s49 s50
*** Adding links:
(h1, s1) (h2, s2) (h3, s3) (h4, s4) (h5, s5) (h6, s6) (h7, s7) (h8, s8) (h9, s9) (h10, s10) (h11, s11) (h12, s12)
(h13, s13) (h14, s14) (h15, s15) (h16, s16) (h17, s17) (h18, s18) (h19, s19) (h20, s20) (h21, s21) (h22,
s22) (h23, s23) (h24, s24) (h25, s25) (h26, s26) (h27, s27) (h28, s28) (h29, s29) (h30, s30) (h31, s31)
(h32, s32) (h33, s33) (h34, s34) (h35, s35) (h36, s36) (h37, s37) (h38, s38) (h39, s39) (h40, s40) (h41,
s41) (h42, s42) (h43, s43) (h44, s44) (h45, s45) (h46, s46) (h47, s47) (h48, s48) (h49, s49) (h50, s50) (s2,
s1) (s3, s2) (s4, s3) (s5, s4) (s6, s5) (s7, s6) (s8, s7) (s9, s8) (s10, s9) (s11, s10) (s12, s11) (s13, s12) (s14,
s13) (s15, s14) (s16, s15) (s17, s16) (s18, s17) (s19, s18) (s20, s19) (s21, s20) (s22, s21) (s23, s22) (s24,
s23) (s25, s24) (s26, s25) (s27, s26) (s28, s27) (s29, s28) (s30, s29) (s31, s30) (s32, s31) (s33, s32) (s34,
s33) (s35, s34) (s36, s35) (s37, s36) (s38, s37) (s39, s38) (s40, s39) (s41, s40) (s42, s41) (s43, s42) (s44,
s43) (s45, s44) (s46, s45) (s47, s46) (s48, s47) (s49, s48) (s50, s49)
*** Configuring hosts
h1 h2 h3 h4 h5 h6 h7 h8 h9 h10 h11 h12 h13 h14 h15 h16 h17 h18 h19 h20 h21 h22 h23 h24 h25 h26 h27
h28 h29 h30 h31 h32 h33 h34 h35 h36 h37 h38 h39 h40 h41 h42 h43 h44 h45 h46 h47 h48 h49 h50
*** Starting controller
c0
*** Starting 50 switches
s1 s2 s3 s4 s5 s6 s7 s8 s9 s10 s11 s12 s13 s14 s15 s16 s17 s18 s19 s20 s21 s22 s23 s24 s25 s26 s27 s28
s29 s30 s31 s32 s33 s34 s35 s36 s37 s38 s39 s40 s41 s42 s43 s44 s45 s46 s47 s48 s49 s50 ...
*** Starting CLI:
mininet&gt;
mininet&gt; h1 ping h50
PING 10.0.0.50 (10.0.0.50) 56(84) bytes of data.
64 bytes from 10.0.0.50: icmp_seq=1 ttl=64 time=6.16 ms
64 bytes from 10.0.0.50: icmp_seq=2 ttl=64 time=0.370 ms
^C
--- 10.0.0.50 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.370/3.268/6.167/2.899 ms
```

Figure 4.92: one topology switch or 50 topology switches

The connection seems to be like this:

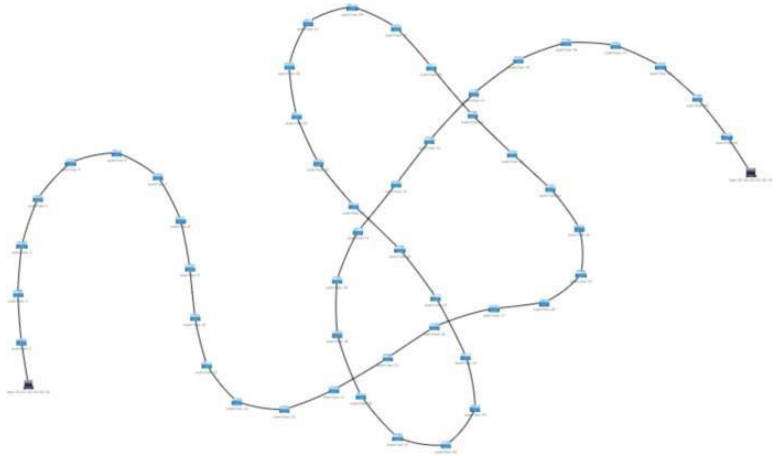


Figure 4.93: Connection of all the host separately on each switch

4.5 Customization of IPv6 Packets

The legacy of the VPN structure and various other protocols make it difficult to synchronize all protocols and standards to simultaneously configure and handle, which is why data packet customization is the key to implementing and testing the proposed framework.

The main customization factors are:

1. The customized values in a data packet carrying the client's identity and other information should be added to the client's traceability, as required by the framework. 1. 1.
2. Testing of botnets required by the system, the numerous hosts operating as botnet wereattempted to attack the infrastructure as DOS/DDOS.
3. Tested initially on IPv4 then integrated more closely with IPv6.
4. The customer can only access privileged applications, and cannot access other networkresources, and the request process can only be sent by the system, and as soon as the system becomes a part of the network. However, customers may use some third-party

   applications to disguise their identity. However, once customers log into that framework, the request processed by their operating system is then customized to packages to gather customized information from the customer's side and track their entire system and network details.

Structure

IPv6 consists of different components and is a very complex thing. Initially, let us explain thecore of IP, the basic IPv6 header, and the extension headers, to give this guide a certain structure. It is required to provide a link for each feature to the RFC where that function is specified and the header structures ("ascii-art" tables from that RFC) are displayed. It also explains that how spacious maps of class members' header fields are.

Basic Scapy Usage

Scapy is a python-written tool that allows easy network packets to be created, manipulated, sent, and received. Perhaps the reader has a fundamental understanding of the spaces and howto use them, so it has not been mentioned into too much detail.

But it is required to describe some of the most important features:

- Different protocols and header types, like IPv6 or ICMP, are shown.
- It can be "stack," just like a real network stack, the protocols on top of each other viaSlash Operation
- (/): IPv6()/TCP() creates a TCP-over-IPv6-Packet.
- The show() and show2() functions can display packets. The send(), sendp(), sr(), sr1(),srp() functions can be used.
    - Manipulate most, if not all, header options. They are class members and can either beaccessed in the builder or by normal membership access, as follows:

```
x=IP(ttl=64)

x.src="127.0.0.1"
```

Figure 4.94: Variable declairation

4.5.1 THE STANDARD IPV6 HEADER

The IPv6 header in RFC2460 (“IPv6”, 1998) is defined and looks as follows:

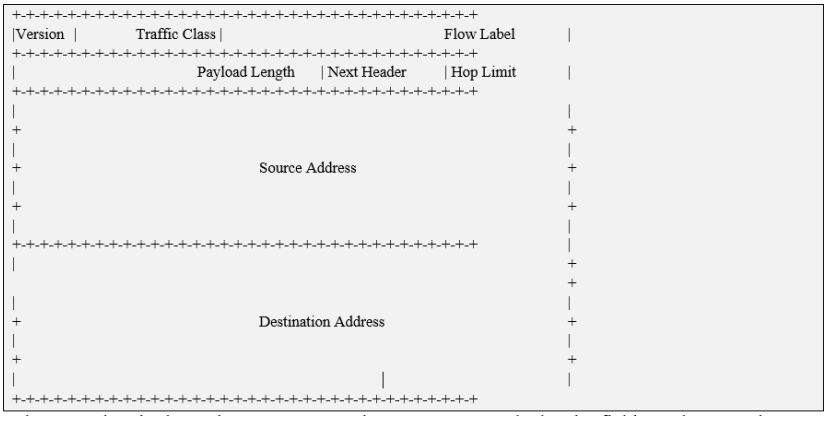


Figure 4.95: IPv6 Header

The IPv6 class is shown in scapy IPv6 packets. Scapy maps the header fields to class membersas follows:

```
>>> ls(IPv6)
Version:    BitField                = (6)
tc          :       BitField        = (0)
fl          :       BitField        = (0)
plen        :       ShortField      = (None)
nh          :       ByteEnumField   = (59)
hlim        :       ByteField       = (64)
src         :       SourceIP6Field = (None)
dst         :       IP6Field        = ('::1')
```

Figure 4.96: Scapy IPv6 packets

Please note in brackets the default values. The "version" field should, for example, have IPv6 with the value "6." If passed one of the send() functions, empty fields will be filled with scapy.The options can be manipulated in the builder or via normal member access.

```
x = IPv6(src='fe80::0123:4567')
x.hlim=255
```

Figure 4.97: IPv6 variable declaration

4.5.2 THE IPV6 EXTENSION HEADERS

The IPv6 header has doubled in size in comparison with IPv4. To avoid further increasing thesize, the basic header contains only the required fields. In extension headers, optional data mustbe provided.

RFC2460 defines 4 expansion headers: Hop-By-Hop header, destination header, Fragment header, and Routing.

Each header (and also the basic IPv6 header) has a next header field. It contains a number specifying the header. The following are:

- Hop-By-Hop: 0
- Routing: 43
- Fragment: 44
- Destination: 60

When the next header is included in another protocol (i.e. TCP or UDP), its required to use itsprotocol numbers (6 or 17, respectively). If the next header does not follow, i.e. a payload-freeIPv6 packet, then the next header should be 59

4.5.3 The Hop-By-Hop Extension Header

The Hop-By-Hop header of IPv6ExtHdrHopByHop-class is represented and looks as follows:

The class members are:

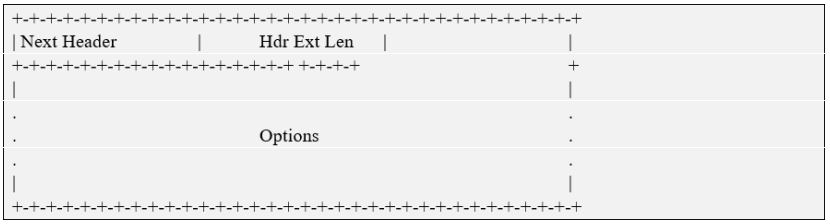


Figure 4.98: The Hop-By-Hop Extension Header

```
>>> ls(IPv6ExtHdrHopByHop)
nh          :        ByteEnumField              = (59)
len         :        FieldLenField              = (None)
autopad     :        _PhantomAutoPadField       = (1)
options :        _HopByHopOptionsField      = ([])
```

Figure 4.99: class members

4.5.4 The Destination Extension Header

The Destination Header is represented by the 'IPv6ExtHdrDestOpt ' 'class:py:class' class andlooks like that:

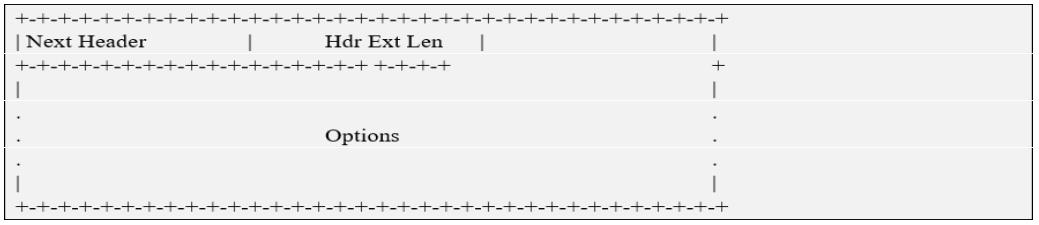


Figure 4.100: Destination Extension Header

The class members are:

```
>>> ls(IPv6ExtHdrDestOpt)
nh : ByteEnumField = (59)
len : FieldLenField = (None)
autopad : _PhantomAutoPadField = (1)
options : _HopByHopOptionsField = ([])
```

Figure 4.101: Destination Class members

4.5.5 The Routing Extension Header

The Routing header is represented by the IPv6ExtHdrRouting-class and looks like this:

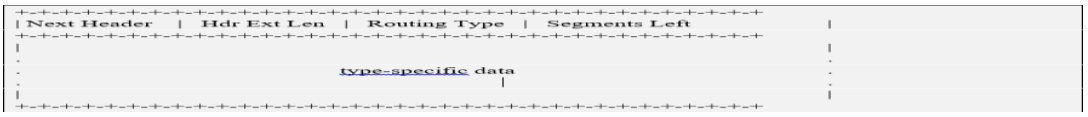


Figure 4.102: he Routing Extension Header

There is currently only the Routing Header Type 0. It is deprecated and should not be

Figure 4.103: Routing Header Type 0

used (seeRFC5095). The type-specific data for RH Type 0 looks like this:

Although deprecated, the RH Type 0 header can still be built with space. The members of thisclass are:

```
>>> ls(IPv6ExtHdrRouting)
nh          : ByteEnumField      = (59)
len         : FieldLenField      = (None)
type        : ByteField          = (0)
segleft     : ByteField          = (None)
reserved    : BitField           = (0)
addresses   : IP6ListField       = ([])
```

Figure 4.104: Class Members

4.5.6 The Fragment Extension Header

The fragment header of the IPv6ExtHdrFragment-class is represented and looks as follows:

The class members are:

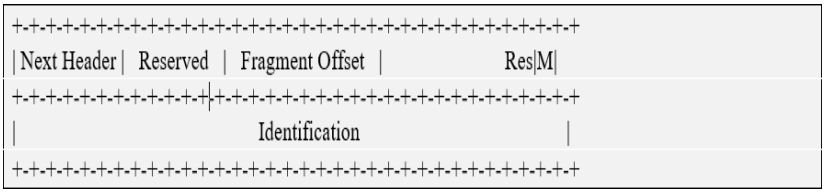


Figure 4.105: 4.5.6The Fragment Extension Header

```
>>> ls(IPv6ExtHdrFragment)
nh : ByteEnumField = (59)
res1 : BitField = (0)
offset : BitField = (0)
res2 : BitField = (0)
m : BitField = (0)
id : IntField = (None)
```

Figure 4.106: Class Members

Note: the res1 and res2 fields are now reserved and unused. The offset is an unsigned integer that measures 8 bytes in the offset. m = 0 is the last fragment; m = 1 means more fragments aregoing to follow. The id identifies the original packet divided into several fragments.

4.5.7 Variable Options

A variable number of options can also be used in the "options"-field of the hop by hop and thedestination header.

The RFC 2460 (IPv6, 1998) was not explicitly defined when it was originally specified. It hasbeen simply stated that it is necessary to have both headers, but not what options. Over the years there have been some suggestions and here is a list of officially specified options.

All headers of options have the format below:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
| Option Type   |   Opt Data Len   |      Option Data
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
```

Figure 4.107: All header options

In the 8-bit Option Type, the Options Data Length specifies the option length and the data option. The option Data is the option. As more than one option can be found in one Hop-By Hop or Destination Header, it may need to adjust options to allow each new option to align naturally and for the whole header to have a length of more than 8 octets. It can be use the Pad1and PadN options for this purpose. See RFC2460 for further information. Scapy can automatically set up the padding.

Scapy currently supports the following:

```
>>> ls(Pad1)
otype              : _OTypeField              = (0)

>>> ls(PadN)
otype              : _OTypeField              = (1)
optlen             : FieldLenField    = (None)
optdata    : StrLenField              = ('')

>>> ls(RouterAlert)
otype              : _OTypeField      = (5)
optlen             : ByteField                = (2)
value              : ShortEnumField   = (None)

>>> ls(Jumbo)
otype              : _OTypeField      = (194)
optlen             : ByteField                = (4)
jumboplen          : IntField         = (None)

>>> ls(HAO)
otype              : _OTypeField      = (201)
optlen             : ByteField                = (16)
hoa                : IP6Field         = ('::')
```

Figure 4.108: scapy to display header options

Not within this guide is to explains all options of the headers. Links to relevant RFCs are provided in the official list.

Some options for which scapy has no distinct classes are available. These include Tunnel Encapsulation Limit, Quickstart, CALIPSO, and RPL-Option. If required to use those features, it's better to create the hex-string and pass to IPv6ExtHdrHopByHop.options.

4.5.8 Examples

In this instance, lets send a spoofed source IPv6 Jumbogram.

- How to create and change an IPv6 packet,
- How to add an Extension Header
- how to operate the variable extension header options.

Since this is the first example of the guide, it will be a little detailed.

Step 1:

The IPv6 packet is created and edited. Enter the destination address and the spoofed sourceaddress as it is not required by scapy to use our actual address automatically:

```
base = IPv6()
base.dst = 'fe80::1234'
base.src = 'fe80::dead:beef'
```

Figure 4.109: The destination address and the spoofed source address

Let scapy figure out all the other settings.

Step 2:

Lets create an Extension Header. The Jumbogram-Option needs to go into the Hop-By-HopHeader:

```
extension = IPv6ExtHdrHopByHop()

jumbo = Jumbo()
```

Figure 4.110: Create Extension Header

Let's now see the Jumbogram. Now. The payload length of the jumbogram is determined by amaximum of 32 bit.

Simply select a large number, like 230 in this example, and paste the jumbo option in the hop-by-hop header.

```
jumbo.jumboplen = 230

extension.options = jumbo
```

Figure 4.111: Jumbogram settings

Step 3:

Stack the headers, inspect the result and pass them to the send function:

```
packet = base/extension
packet.show2()
###[ IPv6 ]###
version= 6L
tc= 0L
fl= 0L
plen= 8
nh= Hop-by-Hop Option Header
hlim= 64
src= fe80::dead:beef
dst= fe80::1234
###[ IPv6 Extension Header - Hop-by-Hop Options Header ]###
nh= No Next Header
len= 0
autopad= On
\options\
|###[ Jumbo Payload ]###
| otype= Jumbo Payload [11: discard+ICMP not mcast, 0: Don't change en-route]
| optlen= 4
| jumboplen= 1073741824
send(packet)
```

Figure 4.112: Implement send function

Done!

4.6 Implementation of End-to-End Encryption

It been a wonder that how WhatsApp keeps chats safe so that the "man in the middle" does notview them. It is called 'End-to-End Encryption' to encode the message and decipher the message after it's received by the recipient before sending a key.

4.6.1 Platforms used:

- Python

4.6.2 Python Packages to be downloaded:

- urllib
- cryptography

To download these, first, install Python. Go to ‘Command Prompt’.Type pip install <package name> and press Enter.

For example:

```
pip install urllib
```

Figure 4.113: install urllib library

Now, here’s the code that sends the message:

```
Import sys
import urllib
import urllib.request
from cryptography.fernet import Fernet
import base64
from cryptography.hazmat.backends import default_backend
from cryptography.hazmat.primitives import hashes
from cryptography.hazmat.primitives.kdf.pbkdf2 import PBKDF2HMAC
msg=str(input('Enter your message : '))
password_provided = input("Provide a key : ")
password = password_provided.encode()
salt = b'salt_'
kdf = PBKDF2HMAC(
algorithm=hashes.SHA256(),
length=32,
salt=salt,
iterations=100000,
backend=default_backend()
)
```

```
key = base64.urlsafe_b64encode(kdf.derive(password))
urllib.request.urlopen
msg=msg.encode()
f = Fernet(key)
msg=f.encrypt(msg)
msg=str(msg)
print("\nYour encrypted text is: "+msg)
b=urllib.request.urlopen('https://api.example.com/update?api_key=489HCQ5PQEJDRFUV&field1='+ms
g)
print("\nYour message has successfully been sent with end-to-end encryption!\nThe receiver needs to
enter the same key.")
```

Figure 4.114: Code that sends the message

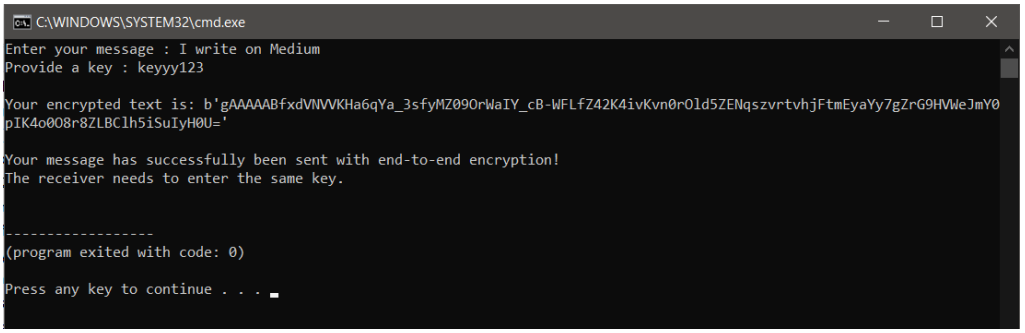


Figure 4.115: Output of Encrypted Message

Type a message now and pass the key. The message is encrypted on a key basis and sent by akey in the code to the cloud channel.

In line 27 of the code at the end of the URL, the data is forwarded to the cloud when it attachesa number or string and opens it.

The code receiving the message is as follows:

```
import csv
import urllib.request
import os
from cryptography.fernet import Fernet
import base64
from cryptography.hazmat.backends import default_backend
from cryptography.hazmat.primitives import hashes
from cryptography.hazmat.primitives.kdf.pbkdf2 import PBKDF2HMAC
url = 'https://example.com/first.csv'
urllib.request.urlretrieve(url, '/Users/YourName/Desktop/first.csv')
```

```
with open('first.csv') as csv_file:
        csv_reader = csv.reader(csv_file)
        for row in csv_reader:
                msg=row[2]
os.remove('/Users/YourName/Desktop/first.csv')
password_provided = input("Provide the key : ")
password = password_provided.encode()
salt = b'salt_'
kdf = PBKDF2HMAC(
        algorithm=hashes.SHA256(),
        length=32,
        salt=salt,
        iterations=100000,
        backend=default_backend()
)
key = base64.urlsafe_b64encode(kdf.derive(password))
f=Fernet(key)
print("\nEncrypted text you've received :\n\n"+msg)
msg=msg[2:-1]
msg=bytes(msg,'utf-8')
msg=f.decrypt(msg)
```

Figure 4.116: Code for the receiver side

Figure 4.117: Output on the receiver side

At the time of message send, it is required to pass the same key. Finally it generate the ciphertext.

Upload the entire history of the input logs in a.csv file on the cloud. Luckily, every time it's thesame URL download. The csv file with first.csv's name is downloaded on execution. It is sentto the file, reads the last value, and then deletes system's file.

What if an incorrect key passed by the user? The message is not displayed.

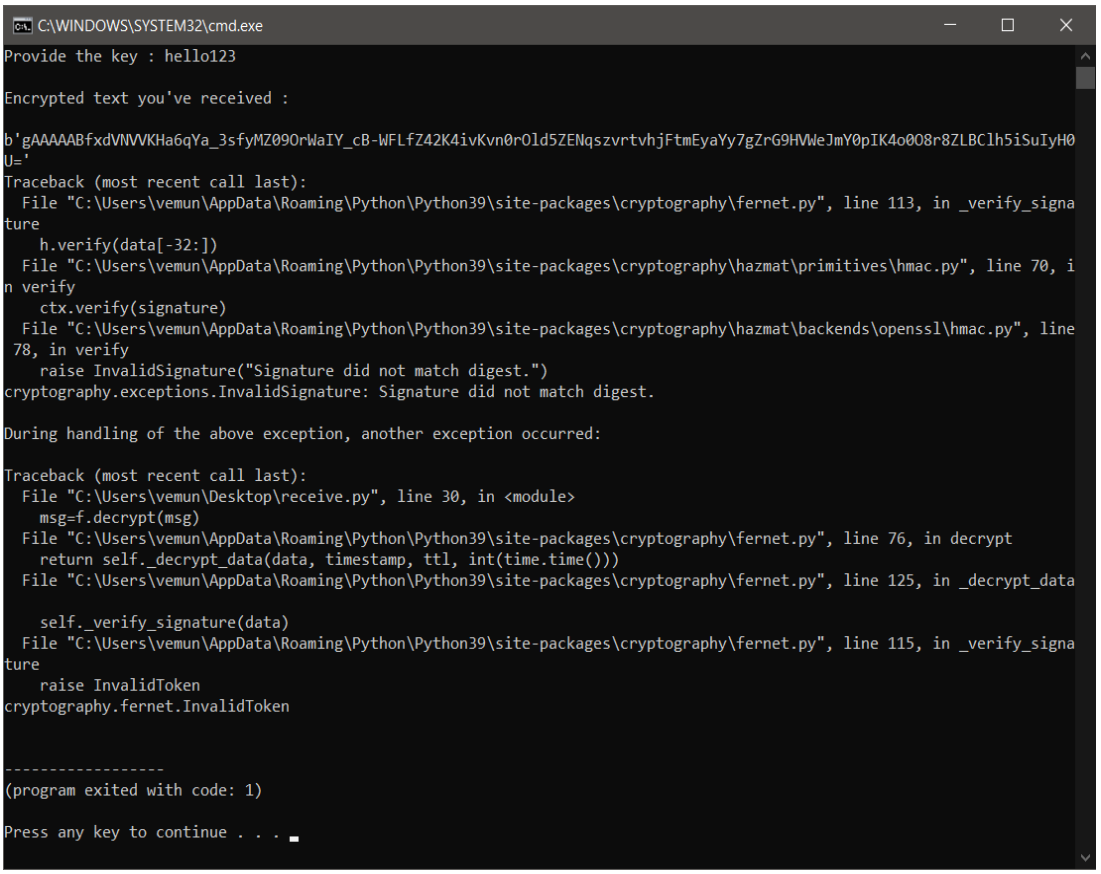


Figure 4.118: Output in case of incorrect key

In case of the ‘Man in the Middle’ attack — The csv file looks like:

created_at	entry_id	field1
2020-10-30 08:24:28 UTC	1	b'gAAAAABfm826qjcHZGO80jprgykmXWHDxwfsVVhx0NWBC5rNnJlph Zd-GBVpH0u_Sl2BmrULF0gNS-9KG-wUAbmrve6Hxzi5_w=='
2020-10-30 08:37:00 UTC	2	b'gAAAAABfm9CqZCdY3Z5mvlM76EpCI999KiD9OV--9RRLgtGwGBkTyzHPurbqlDv7nhuBKDXnDAwX3bZC5dWB5Q7y4P9me 92Grw=='
2020-10-30 08:47:46 UTC	3	b'gAAAAABfm9MwYBnFW3-UVpFe8EbLnvkzbuzTkf1TpcYnmQvS9m5wOhI4krNivmNGGCZ-t6mMHKFzV3-99UDq_Xsa01Pouu9Fog=='
2020-10-30 08:53:36 UTC	4	b'gAAAAABfm9SOPG6bWatP7REFa91O6tYRxlxF14qdExG_a4c-b6EvjhV6FLEbLhX7TdH3V5RXNxXQbaB0hyjeMNI6801bz-rzHQ=='

Table 4.1: In the case of the ‘Man in the Middle attack

4.7 Two Factor Authentication

Use a 2factor authentication (2FA) method to protect accounts if the organization is serious about web security. A combination of password + fingerprint for example is one of the different2FA methods available there. Since not many people have a fingerprint reading

system always available, one of the most popular 2FA methods today is to use a temporary password that expires within a minute (or even less) to create an authenticator app on the cellphone. But how does it work, and how can I implement that on my service? This temporary,time-based, single-time password (TOTP)?

4.7.1 An abstract view

Such authentication is not difficult. The basic requirement is to release a secret key on the server or any other device (usually QR code), and then to generate the passwords via this secretkey. That is why it works, even if the phone is offline since the secret key is stored in the phoneso that it can generate a TOTP perfectly.

4.7.2 Generating the shared secret key

The TOTP algorithm is set in the IETF RFC 6238, which states that the shared key "must be selected by random means or with the use of a well-seeded cryptographically powerfulpseudorandom generator." This key must be encrypted to securely store and should only be decrypted on two occasions: when a password is validated and it must be encrypted when exposed by another device. Lets us use Python to generate the secrets key.

```
import secrets
def generate_shared_secret() -> str:
    return secrets.token_hex(16)
    # >> e8fb1a2faf331bfffe8670ca20447fae
```

Figure 4.119: Generate Shared secret key

Note: This secret key should be unique for all the users in the database, it also guarantees thatone user cannot build a TOTP for another.

4.7.3 Generating and validating a one-time password

Let us create an OTP and validate it, based on a common secret key. Totp = HOTP (K, T), where K is the key that has been just produced, and T, is a time step. This is a basic formula. In other words, that encrypt the timing with our common secret key, but it would not work fora raw timestamp, because the password will be zero for the user to read or input. Therefore, a "step" factor is used so that the user takes a longer time. In the event of usability and safety constraints, RFC 6238 recommends a step of 30 Seconds.

```
import hashlib
import hmac
import math
import time
def generate_totp(shared_key: str, length: int = 6) -> str:
    now_in_seconds = math.floor(time.time())
    step_in_seconds = 30
    t = math.floor(now_in_seconds / step_in_seconds)
    hash = hmac.new(
        bytes(shared_key, encoding="utf-8"),
        t.to_bytes(length=8, byteorder="big"),
        hashlib.sha256,
    )
    return dynamic_truncation(hash, length)
```

Figure 4.120: Generate 2FA, two-factor authentication Secret key

It is required to simply return the HMAC hash, but the user can type the output for too long (even more when there are only 30 seconds to do this). That's what used to get an example of this, usually six digits, with a dynamic truncation algorithm. It was designed by RFC 4226 andconsists of four steps for the predecessor of TOTP:

1 Convert the hash to a binary (base 16) string (base 2)

2 Receive the final four-bit as a whole (base 10)

3 Use this integer as an offset to get the following 32-bit binary string.

4 Turn this 32-bits to integer and obtain the final X-digits where X is the length used

```
def dynamic_truncation(raw_key: hmac.HMAC, length: int) -> str:
    bitstring = bin(int(raw_key.hexdigest(), base=16))
    # >>
110101000001100111010101000100011001000111110010101110100010101101100000101111010001
010111101111110101111011000111010011111000010111111011000010111101111001110011111000
000100010011001010110101111100010111001001000010000011000000010010111100110101100011
    last_four_bits = bitstring[-4:]
    # >> 0011
    offset = int(last_four_bits, base=2)
    # >> 3
    chosen_32_bits = bitstring[offset * 8 : offset * 8 + 32]
    # >> 01000100011001000111110010101110
    full_totp = str(int(chosen_32_bits, base=2))
    # >> 1147436206
    return full_totp[-length:]
    # >> 436206
```

Figure 4.121: Turn 32 bits to integer

For this example, 436206 is the user's temporary password. Now, to check the backend for a password, it’s the same thing. Use the shared key to generate a TOTP on the server and checkwhether it matches the information.

```
def validate_totp(totp: str, shared_key: str) -> bool:

    return totp == generate_totp(shared_key)
```

Figure 4.122: Function to match the information

4.7.4 Conclusion

It is not difficult to implement 2FA but must be taken seriously to avoid infringements. Thereis no perfect security protocol, no silver bullet, but why not make life more difficult for an invader.

4.8 HTTPS (SSL) packets on VPN-IPSEC

Tunneling Traffic over TLS VPN Bypassing Deep Packet Inspection

Network operators in some countries use deep packet inspection methods to block certain traffic types. For users to avoid sending encrypted packets over such reseals, Virtual Private Network (VPN) traffic can be analyzed and blocked.

When it has been observed that HTTPS works throughout the world and cannot be easily analyzed (the payload is generally encrypted), the argument goes that VPN tunnels can also beorganized in the same way: It is possible to construct a secure and reliable network by disclosing the VPN traffic with TLS or its older SSL version. Packets sent via such tunnels arepermitted to cross multiple domains that have different security policies (strict and not so strict). Whilst the SSH could potentially be used to create such networks, there is evidence thatconnections made via such tunnels in certain countries are being analyzed based on statistics: when the network use is high, there are explosions or connections which live for a long time, the network operators are resetting the underlying TCP connections.

So here the experimental effort took place: The first is to describe the different VPN solutionsavailable on the Internet; the second is to describe our experimental effort with Python-based sound and Linux software that enables users to create VPN tunnels via the TLS protocol and SOHO Tunnel.

4.8.1 INTRODUCTION

The modern era is dominated by virtual private networks (VPNs). Users can obtain network services, which would otherwise be blocked by network operators, by encapsulating and sending customer traffic into protected tunnels. VPN solutions also help with the intranet network of a company. Corporate employees, for example, can securely access the internal network via VPN and direct all transmission through the

tunnel to the corporate network. Theycan obtain services that would otherwise be impossible to obtain from outside the world.

4.8.2 BACKGROUND

There are different solutions for building VPNs. Host Identity Protocols (HIP) is one example(Gurtov, 2008). The HIP is a layer 3 solution that was originally designed to split the dual role ofthe IP addresses, i.e. the identifier and locator, between transport and network layers. For example, a Tempered Networks company uses HIP protocol to construct safe networks (for sampling see (Musthaler, 2019)).

Secure Shell protocol is another solution (or SSH). SSH is a protocol for the application layerthat supplies an encrypted channel for unsafe networks. Originally, SSH was designed to secureremote control lines, login, and execution of commands (SSH, 2020). SSH can secure any network service. In addition, SSH provides means to create VPN tunnels between spatially isolated networks. Sadly, it is possible to analyze and block SSH connection (would it be as widely spread for example TLS protocol, things could be different).

Like SSH, the TCP protocol is run by OpenVPN (OpenVPN, 2020) (in fact, OpenVPN can also operate on top of UDP transport protocol). It has been proofed that the successful blockingof OpenVPN by governments in certain countries. Of course, the detection of these protocols is more difficult, since traffic is enclosed within TCP/UDP connections. To block such tunnelsefficiently, deep packet inspection solutions are required.

The IPSec protocol (IPsec, 2020) is another widely used layer 3 protocol to build VPNs. It is possible to create an IPSec security association by pre-shared keys or by using protocols of Internet Key Exchange (IKE and IKEv2) (IKE, 2020). Since IPsec runs directly above IP protocol, without the use of complex packet inspection solutions it can be easily detected.

4.8.3 HARDWARE, SOFTWARE, AND ARCHITECTURE

The general view of architecture is shown in Figure 1. Please note, it has been considered thata client SOHO VPN box is a separate, path-not-internal Linux box.

It was decided to use the Python framework and Ubuntu Linux distribution to implement the VPN client and server. The implementation consists of around 1.2K code lines (LOC) and all functions are implemented in a user environment.

However, the following configuration was considered during the experiments. The opted cloudhardware is from a DigitalOcean on a micro-instance. The instance was installed in New York,USA, using a single CPU, 25 GB storage, 1 GB of random access memory. The VPN client was based on a Raspberry PI microcontroller in Germany. The SOHO router imitated on this way. It has also been decided to use a tool to measure performance and use ping tool to measureround trip times.

4.8.4 EXPERIMENTAL EVALUATION

Throughout this research work, several experiments have been conducted. The first experimentwas to send ping messages to the Google DNS server and to observe the differences during thejourney (basically, it has been compared the round trip times between the setting in which the tunnel was present and the setting in which the tunnel did not exist).

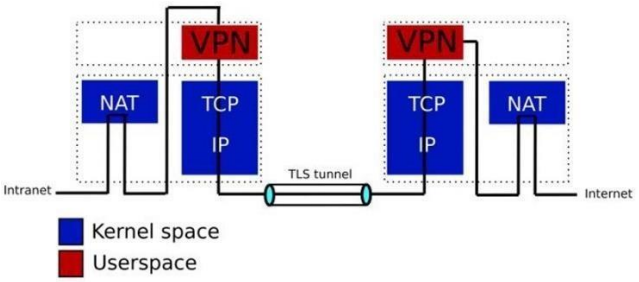


Figure 4.123: TLS tunnel between Intranet and Internet

Next, lets identify to measure the transmission between the local and distant machines. In essence, carried out 50 measurements for a position where the traffic was entering the tunnel and for the setting in which the traffic was normal operating (meaning, unencrypted and not encapsulated in TLS packets). The utility “wget” used to download Linux kernel file1 (1.3 MB)from the Web to measure the performance. Figure 4.124 illustrates the round trip distribution times (RTT). Mean RTT was 293.7 ms, andmean RTT was 103.3 ms for a single ICMP.

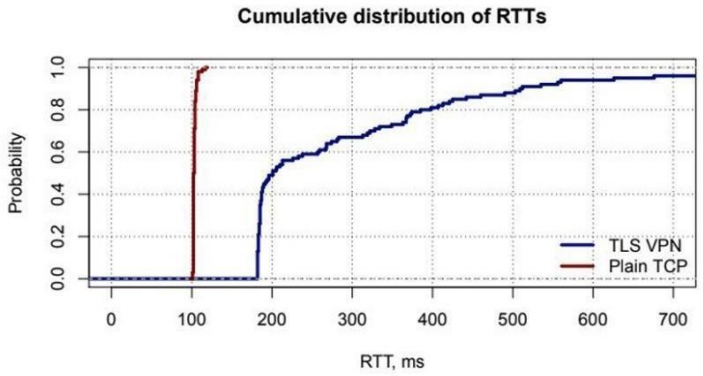


Figure 4.124: Cumulative distribution of RTT's

The distribution of the obtained throughput for both TLS protected tunnels and regular TCP links are presented in Figure 4.124. The medium VPN connection throughput value was 608.7Kb/s, and the average TCP connection throughput was 1890.4 Kb/s. Given these findings, its

believed that our implementation is the bottleneck in this case. However, the VPN tunnel implementation for smaller office settings is sufficient.

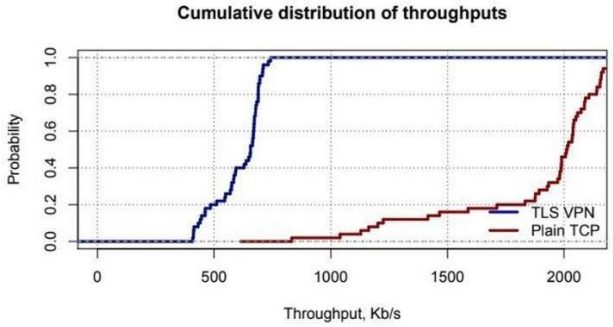


Figure 4.124: Cumulative distribution of throughputs

4.8.5 POSSIBLE EXTENSION

Although the tunnel was running continuously for several hours and implemented the authentication layer, there were still several connectivity attempts from outside the world, though unsuccessful. Let's hide the VPN server from the web server in the future so that it canonly be accessed on the VPN server by those who know some secret. The idea is easy: if the user calls a secret page with HTTP GET requests, the webserver opens a port and redirects traffic from the client to the VPN server. The server closes the port after the connection has been established.

4.8.6 CONCLUSIONS

In this short report, It has been tried to describe that how network operators can block VPN solutions. It has been argued that VPN traffic can still be concealed from observers, using TLSprotocol to mask tunnels. Experiments suggest that when a secure channel is required, this approach is useful, but network operators are keen to block VPN traffic.

While it is difficult to do business with this solution (network operators can simply block all IP addresses of a business-hosted VPN server, in the long term), Its maintained that people can usethis software and use it on their cloud machines (they can keep the IP addresses of the servers in secret). This means that network operators will have little chance of identifying the server'sIP addresses: It is a pretty complex task for network operators to scan all IP addresses and findthose that are used to send VPN traffic so that traffic is as normal HTTPS (at the end of the tunnel is the SSL using standard HTTS port), or as a pretty complex task.

4.9 Log Analysis:

It is a general framework that identifies problems with system logs. It uses system logs and KPI metrics to identify impacting system problems quickly and accurately. There are four steps:

1. Parsing of the logs
    - **General Framework**
2. Vectorization sequence
3. Cascading Clustering and
4. Analysis of correlation.

The core component is the cascading clustering algorithm, which groups numerous sequentialvectors by iteratively sampling, clustering, and matching.

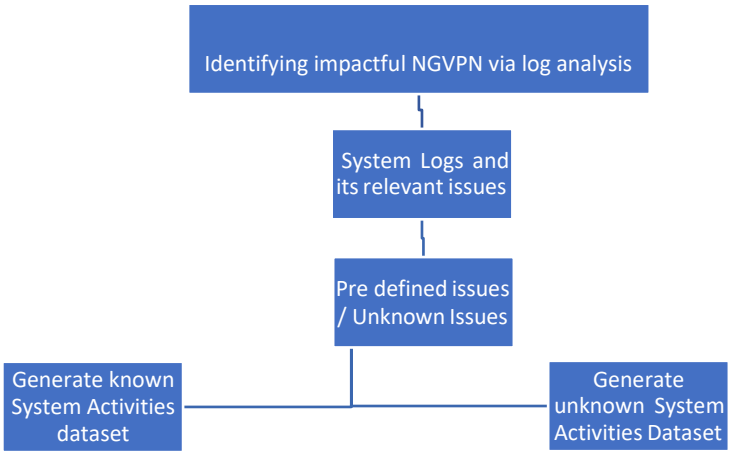


Figure 4.125: Log Analysis

Structure:

1. run.py: Main entry function that defines the hyper-parameters required.

5 cascading_clustering.py: Cascading clustering algorithm implementation.

6 dataloader.py: load the input data files into memory

7 save_results.py: save the clustering result into files.

Data Format:

Multiple log-sequence files: within a time interval, each of these files consists of log-sequencevectors.

The KPI data: Every value of the KPI is equivalent to the time interval system status.

2. The Time interval log sequence matrix:

Log Seq	Event 1	Event 2	Event 3	Event 4	...
1	2	1	0	2	...
2	3	2	4	1	...
3	2	1	3	3	...
...	...	...	...	...	...

Table 4.2: Time interval log sequence matrix

Assume that the total time intervals of N are, then N matrices and N KPI values are available.These K KPI values are stored in the following file.

1. KPI data

Time Interval	KPI
T0	0.05
T1	0.10
T2	0.07
..	...

Table 4.3: KPI data

The N KPI values contain in the KPI file.

Code:

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
import cascading_clustering as cluster
import time
#    Replace the parameters in the following "para" according to your data
if __name__ == '__main__':
        para = {
                'seq_folder':  'seq_folder/',    # folder of log sequence matrix file
                'kpi_path':    'kpi_path',       # the path of KPI file
                'proc_num':    16,               # number of processes when loading files and saving files
                'sample_rate': 100,              # sample rate for sampling, 100 represents 1% sample rate
                'thre':        0.3,              # threshold for clustering, and also used when matching the
nearest sequence
                'saveFile':    False,            # FLAG to decide whether saving output clusters, it costs a lot
if turned on, default False
                'output_path': 'output_path',    # folder for saving output clusters of data
                'rep_path':    'rep_path/',      # path used for savinng all representatives (patterns).
                }
        kpipath = para['kpi_path']
        t1 = time.time()
        rawData, rawIndex, eveOccuMat = cluster.loading_all_data(para)
        kpiList = cluster.load_kpi(kpipath)
        corWeightList = cluster.get_corr_weight(eveOccuMat, kpiList)
        weight_data, weightList = cluster.weighting(rawData, corWeightList)
        finacluResult = cluster.cascading(para, rawData, rawIndex, weight_data)
        t2 = time.time() - t1
        print('the entire time usage is ', t2)
```

Figure 4.126: Log Analysis Code

4.9.1 Analyze log data with Python and Apache Spark

By using Python and Apache Spark, our weblogs are processed and tamed into an analytical format that is essential given the huge amount of log data that are generated today by most companies. To work with both DataFrames and regular expressions, the environment variablesand dependencies were set up. Of course, the example log data were loaded. Then the log datawas rubbed into a clean, structural and meaningful format. Part 2 focuses mainly on the analysisof these data.

Data analysis on weblogs

Now let’s try some interesting Exploratory Data Analysis (EDA) to get interesting insights from a DataFrame that includes the parsed and cleaned log file as data frame rate.

Content size statistics

Let's calculate statistics on the size of content returned by our web server. In particular, theaverage, minimum, and maximum content sizes would be desirable.

Call.describe() to calculate this statistics on the logs df column content size. The function

.describe() returns a column in this format count, mean, stddev, min, and maximum:

```
content_size_summary_df = logs_df.describe(['content_size'])
content_size_summary_df.toPandas()
```

Figure 4.127: Functions for Content size

	summary	content_size
0	count	3461612
1	mean	18928.844398216785
2	stddev	73031.47260949228
3	min	0
4	max	6823936

Table 4.4: Output Content Size

Static analysis of the size of content returned by the webserver.

Alternatively, use SQL to calculate these statistics directly. The function module pyspark.sql.Has many helpful features

Call .agg() toPandas() to extract and convert the results to DataFrame pandas that allow betterJupyter Notebook formatting:

```
from pyspark.sql import functions as F

(logs_df.agg(F.min(logs_df['content_size']).alias('min_content_size'),
             F.max(logs_df['content_size']).alias('max_content_size'),
             F.mean(logs_df['content_size']).alias('mean_content_size'),
             F.stddev(logs_df['content_size']).alias('std_content_size'),
             F.count(logs_df['content_size']).alias('count_content_size'))
        .toPandas())
```

Figure 4.128: Code in Python using Pandas

	min_content_size	max_content_size	mean_content_size	std_content_size	count_content_size
0	0	6823936	18928.844398	73031.472609	3461612

Table 4.5: data transformed into pandas

The same data was transformed into data pandas.

The results are the same as expected based on its validation.

HTTP status code analysis

Let's then take a look at the status values of the log, to see which and how many times status code values appear. Let’s restart logs_df, group the status column, use the aggregationfunction.count() and sort by the column of status:

```
status_freq_df = (logs_df
                  .groupBy('status')
                  .count()
                  .sort('status')
                  .cache())
print('Total distinct HTTP Status Codes:', status_freq_df.count())
```

Figure 4.129: HTTP status code analysis

Total unique HTTP status codes: 8

Let's examine every event in the form of a frequency table in every status code:

```
status_freq_pd_df = (status_freq_df
                     .toPandas()
                     .sort_values(by=['count'],
                                  ascending=False))
status_freq_pd_df
```

Figure 4.130: frequency table in every status code

	status	count
0	200	3100524
2	304	266773
1	302	73070
5	404	20899
4	403	225
6	500	65
7	501	41
3	400	15

Table 4.5: Output of every status code

List of logs containing each status code frequently.

The most common status code seems to be 200—OK—it’s a good sign usually server respondsmost of the time. Let's view this:

```
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
%matplotlib inline

sns.catplot(x='status', y='count', data=status_freq_pd_df,
            kind='bar', order=status_freq_pd_df['status'])
```

Figure 4.131: common status code

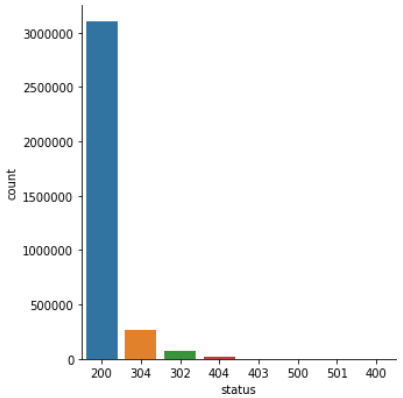


Figure 4.132: HTTP status code occurrences in a bar chart.

Due to the huge skew of the data, several status codes are almost invisible. Let's change a log and see whether things get better. A log usually transforms highly skewed data into an approximate ordinary distribution, so that the distribution of data is visualized more understandably:

```
log_freq_df = status_freq_df.withColumn('log(count)',
                                        F.log(status_freq_df['count']))
log_freq_df.show()
```

Figure 4.133 Status code, count, and log frequency of count

status	count	log(count)
200	3100524	14.947081687429097
302	73070	11.199173164785263
304	266773	12.494153388502301
400	15	2.70805020110221
403	225	5.41610040220442
404	20899	9.947456589918252
500	65	4.174387269895637
501	41	3.713572066704308

Table 4.6: Output of Status code, count, and log frequency of count

Error code frequency as a log transform.

The results are certainly good and the skew appears to have been handled, let us check this byviewing this information:

```
log_freq_pd_df = (log_freq_df
                  .toPandas()
                  .sort_values(by=['log(count)'],
                               ascending=False))
sns.catplot(x='status', y='log(count)', data=log_freq_pd_df,
            kind='bar', order=status_freq_pd_df['status'])
```

Figure 4.134: Error code frequency

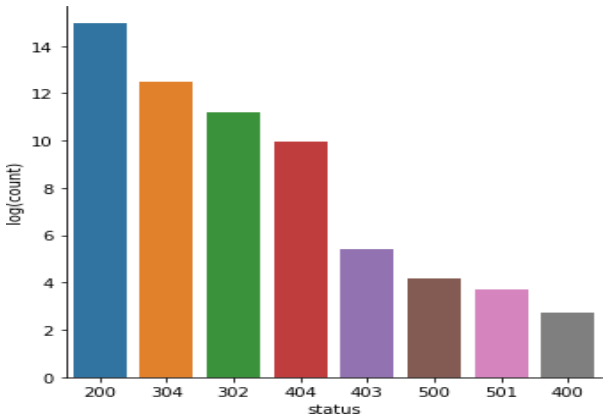


Figure 4.135: HTTP status code frequency bar chart, after a log transform

The chart looks much better and less skewed, which gives us an understanding of how status codes are distributed!

Analyzing frequent hosts

Let the hosts who frequently access the server look, when the total number of accesses is determined by each host, the number of accesses is sorted and only the top ten hosts are displayed:

Figure 4.136: Analyzing frequent hosts

```
host_sum_df =(logs_df
        .groupBy('host')
        .count()
        .sort('count', ascending=False).limit(10))
host_sum_df.show(truncate=False)
```

host	count
sedulitygroups.com	32653
tmv.edu.in	23664
gmail.com	22783
edams.ksc.nasa.gov	12911
2001:ac23:a22f::0	8767

en.wikipedia.org	7349
youtube.com	7109
	5083
aajtak.com	3215

Table 4.7: Hosts which frequently access the number of accesses sorted by server.

The table looks good, but let's take a closer look at the blank record in row 9:

```
host_sum_pd_df = host_sum_df.toPandas()
host_sum_pd_df.iloc[8]['host']
```

Figure 4.137: Check for empty strings as well

Seems like an empty string is one of the top hostnames. This example teaches us a preciouslesson: don't check nulls only for data wrangling, but also for empty strings.

Display the top 20 most frequent endpoints

Let's now display how many URI endpoint hits the log. To do this, start with logs_df and thengroup by the endpoint column, aggregate by count, and sort as follows:

```
paths_df = (logs_df
        .groupBy('endpoint')
        .count()
        .sort('count', ascending=False).limit(20))

paths_pd_df = paths_df.toPandas()
paths_pd_df
```

Figure 4.138: 20 most frequent endpoints

	endpoint	count
0	/images/NASA-logosmall.gif	208714
1	/images/KSC-logosmall.gif	164970
2	/images/MOSAIC-logosmall.gif	127908
3	/images/USA-logosmall.gif	127074
4	/images/WORLD-logosmall.gif	125925
5	/images/ksclogo-medium.gif	121572
6	/ksc.html	83909
7	/images/launch-logo.gif	76006
8	/history/apollo/images/apollo-logo1.gif	68896
9	/shuttle/countdown/	64736
10	/	63171
11	/images/ksclogosmall.gif	61393
12	/shuttle/missions/missions.html	47315
13	/images/launchmedium.gif	40687
14	/htbin/cdt_main.pl	39871
15	/shuttle/missions/sts-69/mission-sts-69.html	31574
16	/shuttle/countdown/liftoff.html	29865
17	/icons/menu.xbm	29190
18	/shuttle/missions/sts-69/sts-69-patch-small.gif	29118
19	/icons/blank.xbm	28852

Table 4.8: indicating in descending order the number of hits at each endpoint URI.

Not surprisingly, GIFs, the home page, and some CGI scripts are the most accessed assets.

Display the top 10 error endpoints

A sorted list of endpoints and number of times a non-200 return code was accessed, and then atop 10 is displayed:

```
not200_df = (logs_df
             .filter(logs_df['status'] != 200))

error_endpoints_freq_df = (not200_df
                           .groupBy('endpoint')
                           .count()
                           .sort('count', ascending=False)
                           .limit(10)
                       )

error_endpoints_freq_df.show(truncate=False)
```

Figure 4.139: Display the top 10 error endpoints

endpoint	count
/images/NASA-logosmall.gif	40082
/images/KSC-logosmall.gif	23763
/images/MOSAIC-logosmall.gif	15245
/images/USA-logosmall.gif	15142
/images/WORLD-logosmall.gif	14773
/images/ksclogo-medium.gif	13559
/images/launch-logo.gif	8806
/history/apollo/images/apollo-logo1.gif	7489
/	6296
/images/ksclogosmall.gif	5669

Table 4.9: The top ten error endpoints and their frequency.

The most loaded looks like GIFs (animated/static images). It is because those logs date back toold servers and about given the Internet speed that was available before.

Total number of unique hosts

In these two months, how many unique hosts did NASA visit? With a few transformations ithas been identified the output:

```
unique_host_count = (logs_df
                     .select('host')
                     .distinct()
                     .count())
unique_host_count


Output:

137933
```

Figure 4.140: Total number of unique hosts

Number of unique daily hosts

Let's see how to determine the number of single hosts daily for an advanced example. The dataframe is required here, which comprises the day of the month and the number of unusual hosts for the day, sorted by the day of the month. Think of how to accomplish this task. Because each log only covers one month, at least the month issue can be ignored. For data that extends over several months, the necessary aggregations must be considered both month and day. Please use the dayofmonth() function ofthe pyspark.sql.functions module (which have been already imported as F at the beginning of this tutorial). Start with host_day_df, which is a DataFrame with two columns:

column	explanation
host	the host name
day	the day of the month

Table 4.10: The columns in the host_day_df data frame

For each row in logs_df, there is a row in this DataFrame. Essentially, each row is just being transformed. In this row, for instance:

```
unicomp6.unicomp.net - - [01/Aug/1995:00:35:41 -0400] "GET /shuttle/missions/sts-73/news
HTTP/1.0" 302 -|
```

Figure 4.141: Each row transformed in logs

The host_day_df should have unicomp6.unicomp.net

```
host_day_df = logs_df.select(logs_df.host,
                    F.dayofmonth('time').alias('day'))
host_day_df.show(5, truncate=False)
```

Figure 4.142: function host_day_df

```
+--------------------+---+
|host                |day|
+--------------------+---+
|199.72.81.55        |1  |
|unicomp6.unicomp.net|1  |
|199.120.110.21      |1  |
|burger.letters.com  |1  |
|199.120.110.21      |1  |
+--------------------+---+
only showing top 5 rows
```

Table 4.11: On the first day, the top five hosts make requests.

Another option, a DataFrame with two-column, which is different than the previous DataFrames, is daily_unique_hosts_df:

column	explanation
day	the day of the month
count	the number of unique requesting hosts for that day

Table 4.12: Columns shown by daily_unique_hosts_df

```
def_mr = pd.get_option('max_rows')
pd.set_option('max_rows', 10)

daily_hosts_df = (host_day_distinct_df
                  .groupBy('day')
                  .count()
                  .sort("day"))

daily_hosts_df = daily_hosts_df.toPandas()
daily_hosts_df
```

Figure 4.143: display day and count

	day	count
0	1	7609
1	2	4858
2	3	10238
3	4	9411
4	5	9640
...	...	...
26	27	6846
27	28	6090
28	29	4825
29	30	5265
30	31	5913

31 rows × 2 columns

Table 4.13: Daily_unique_hosts_df shows the number of single hosts requesting the day in which the month isdisplayed

This result provides us with a nice DataFrame that shows all unique hosts per day. Let usvisualize this:

```
c = sns.catplot(x='day', y='count',
                data=daily_hosts_df,
                kind='point', height=5,
                aspect=1.5)
```

Figure 4.144: shows all unique hosts per day

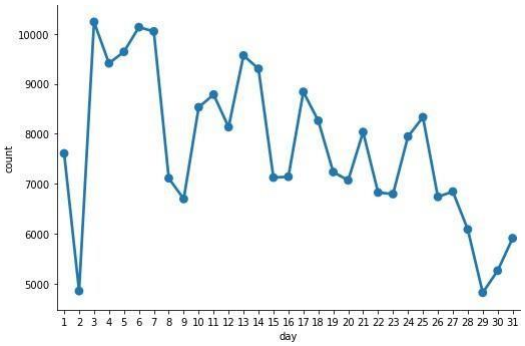


Figure 4.145: Unique hosts per day charted daily_unique_hosts_df.

The average number of daily requests per host

In the last example, let us look at one way to determine daily the number of unique hosts throughout the log. Now let's find the average number of requests for the NASA website per host per day. Here it is required an increasing day of the month-sized data frame that containsa day of the month and the corresponding average number of requests made per host for that day:

```
daily_hosts_df = (host_day_distinct_df
                  .groupBy('day')
                  .count()
                  .select(col("day"),
                          col("count").alias("total_hosts")))

total_daily_reqests_df = (logs_df
                          .select(F.dayofmonth("time")
                                  .alias("day"))
                          .groupBy("day")
                          .count()
                          .select(col("day"),
                                  col("count").alias("total_reqs")))

avg_daily_reqests_per_host_df = total_daily_reqests_df.join(daily_hosts_df, 'day')
avg_daily_reqests_per_host_df = (avg_daily_reqests_per_host_df
                                 .withColumn('avg_reqs', col('total_reqs') / col('total_hosts'))
                                 .sort("day"))
avg_daily_reqests_per_host_df = avg_daily_reqests_per_host_df.toPandas()
avg_daily_reqests_per_host_df
```

Figure 4.146: Average number of daily requests per host

	day	total_reqs	total_hosts	avg_reqs
0	1	98710	7609	12.972795
1	2	60265	4858	12.405311
2	3	130972	10238	12.792733
3	4	130009	9411	13.814579
4	5	126468	9640	13.119087
...	...	...	...	...
26	27	94503	6846	13.804119
27	28	82617	6090	13.566010
28	29	67988	4825	14.090777
29	30	80641	5265	15.316429
30	31	90125	5913	15.241840

31 rows × 4 columns

Table 4.14: The average daily number of host requests by avg_daily_requests_per_host_df.

The average daily request can now be displayed per host:

```
c = sns.catplot(x='day', y='avg_reqs',
                data=avg_daily_reqests_per_host_df,
                kind='point', height=5, aspect=1.5)
```

Figure 4.147: Average daily request

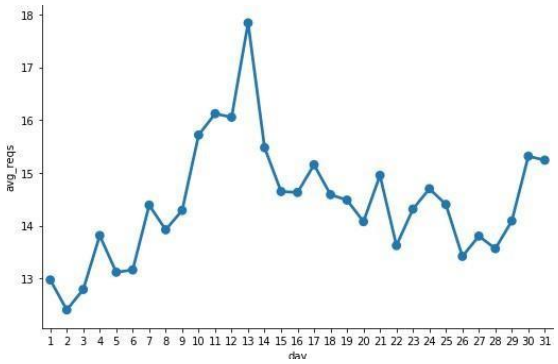


Figure 4.148: The average daily request number per host is calculated.

The maximum number of requests per host seems to be day 13.

Counting 404 response codes

Creates a Dataframe with a 404 status code only for log records (Not Found). The cache()function used to create memory space for large information.

```
not_found_df = logs_df.filter(logs_df["status"] == 404).cache()
print(('Total 404 responses: {}').format(not_found_df.count()))

OUTPUT:

Total 404 responses: 20899
```

Figure 4.149: Counting 404 response codes

Listing the top twenty 404 response code endpoints

With the DataFrame, which has been cached before—with only logs that have a 404 reply code—Let us print a list of the top 20 endpoints that generate 404 errors. Remember, they should be sorted in order when generating top endpoints:

```
endpoints_404_count_df = (not_found_df
                    .groupBy("endpoint")
                    .count()
                    .sort("count", ascending=False)
                    .limit(20))

endpoints_404_count_df.show(truncate=False)
```

Figure 4.150: list of top 20 endpoints that generate 404 errors

endpoint	count
/pub/winvn/readme.txt	2004
/pub/winvn/release.txt	1732
/shuttle/missions/STS-69/mission-STS-69.html	683
/shuttle/missions/sts-68/ksc-upclose.gif	428
/history/apollo/a-001/a-001-patch-small.gif	384
/history/apollo/sa-1/sa-1-patch-small.gif	383
/://spacelink.msfc.nasa.gov	381
/images/crawlerway-logo.gif	374
/elv/DELTA/uncons.htm	372
/history/apollo/pad-abort-test-1/pad-abort-test-1-patch-small.gif	359
/images/nasa-logo.gif	319
/shuttle/resources/orbiters/atlantis.gif	314
/history/apollo/apollo-13.html	304
/shuttle/resources/orbiters/discovery.gif	263
/shuttle/missions/sts-71/images/KSC-95EC-0916.txt	190
/shuttle/resources/orbiters/challenger.gif	170
/shuttle/missions/technology/sts-newsref/stsref-toc.html	158
/history/apollo/images/little-joe.jpg	150
/images/lf-logo.gif	143
/history/apollo/publications/sp-350/sp-350.txt~	140

Table 4.15: With the endpoints_404_count_df, the best 20 response code endpoints are sorted.

Listing the top twenty 404 response code hosts

The previously cached DataFrame, which only contains log records that have 404 response codes, allows us now to print a list of the top twenty hosts with the most 404 bugs. Remember,again, that top hosts in sorted order should be:

```
hosts_404_count_df = (not_found_df
                    .groupBy("host")
                    .count()
                    .sort("count", ascending=False)
                    .limit(20))

hosts_404_count_df.show(truncate=False)
```

Figure 4.151: The top twenty 404 response code hosts

host	count
hoohoo.ncsa.uiuc.edu	251
piweba3y.prodigy.com	157
jbiagioni.npt.nuwc.navy.mil	132
piweba1y.prodigy.com	114
	112
www-d4.proxy.aol.com	91
piweba4y.prodigy.com	86
scooter.pa-x.dec.com	69
www-d1.proxy.aol.com	64
phaelon.ksc.nasa.gov	64
dialip-217.den.mmc.com	62
www-b4.proxy.aol.com	62
www-b3.proxy.aol.com	61
www-a2.proxy.aol.com	60
www-d2.proxy.aol.com	59
piweba2y.prodigy.com	59
alyssa.prodigy.com	56
monarch.eng.buffalo.edu	56
www-b2.proxy.aol.com	53
www-c4.proxy.aol.com	53

Table 4.16: The top 20 404 reply code is provided via hosts_404 _count _df.

This result gives us a good idea of which host generates NASA's website with the most 404errors.

Visualizing 404 errors per day

Let's now temporarily examine our 404 records (by time). Similar to the example that showsthe number of unique hosts per day, lets break up the 404 daily requests and solve daily errorcounts. _in file sorted_df:

```
errors_by_date_sorted_df = (not_found_df
                            .groupBy(F.dayofmonth('time').alias('day'))
                            .count()
                            .sort("day"))

errors_by_date_sorted_pd_df = errors_by_date_sorted_df.toPandas()
errors_by_date_sorted_pd_df
```

Figure 4.152: Visualizing 404 errors per day

	day	count
0	1	559
1	2	291
2	3	778
3	4	705
4	5	733
...	...	...
26	27	706
27	28	504
28	29	420
29	30	571
30	31	526

31 rows × 2 columns

Table 4.17: 404 mistakes a day via date sorted_df_errors.

Let us now visualize the 404 errors a day:

```
c = sns.catplot(x='day', y='count',
                data=errors_by_date_sorted_pd_df,
                kind='point', height=5, aspect=1.5)
```

Figure 4.153: visualize the 404 errors a day

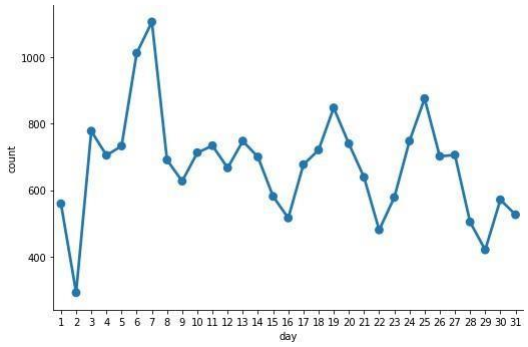


Figure 4.154: Per day Total 404 errors

4.9.2.2 Top three days for 404 errors

Based on the previous plot, which is the most 404 errors during the top three days of the month. Knowing this can help us diagnose and further explore what might have gone wrongin these spectacular days (server issues, DNS issues, denial of service, latency problems, maintenance, and so on). Let us try previous errors by date sorted df DataFrame:

```
(errors_by_date_sorted_df
    .sort("count", ascending=False)
    .show(3))
```

Figure 4.155: Top three days for 404 errors

```
+---+-----+
|day|count|
+---+-----+
|  7| 1107|
|  6| 1013|
| 25|  876|
+---+-----+
only showing top 3 rows
```

Table 4.18: The top 3 days of 404 errors via errors_by_date_sorted_df.

The top 3 days of 404 errors via errors_by_date_sorted_df.

Top three days for 404 errors

With the not found df_dataframe, which was cached before, lets group and sort more and more by an hour of the day. This process will be used to create a data frame with a total of 404 responses per hour of the day for HTTP requests (midnight starts at 0). Further it supportto create a DataFrame visualization.

```
hourly_avg_errors_sorted_df = (not_found_df
                               .groupBy(F.hour('time')
                                        .alias('hour'))
                               .count()
                               .sort('hour'))
hourly_avg_errors_sorted_pd_df = hourly_avg_errors_sorted_df.toPandas()

c = sns.catplot(x='hour', y='count',
                data=hourly_avg_errors_sorted_pd_df,
                kind='bar', height=5, aspect=1.5)
```

Figure 4.156: Top three days for 404 errors

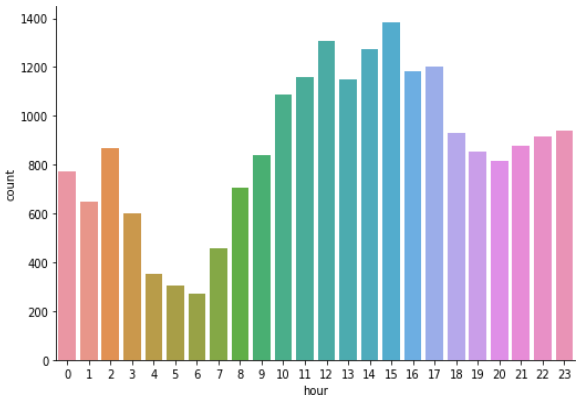


Figure 4.157: Total 404 errors per hour in a bar chart.

It appears that a total of 404 mistakes occur in the afternoon the most and early morning theleast. The pandas can now reset the maximum rows as changed it to display a limited numberof rows earlier.

```
pd.set_option('max_rows', def_mr)
```

Figure 4.158: A total of 404 mistakes occur

CONCLUSION

A practical approach in a very common, but essential, case study on log analysis to the architectural, analytical, and visualization of data on the scale. Although the data collected on which worked upon may not be "large data" in terms of size or volume, the methods and methods are generic enough to scale for larger amounts. It's been hoped that this practice has given lots of ideas on howopen source frames such as Apache Spark can be used to work with structured and semi-structureddata.

Password Policy

Password policies were enforced to every user of the system in other to provide additional level of securityto the Brute force prevention system. These policies must be followed in order to utilize the system else insome situations, the user will be denied access to the system. These policies include:

Every password must contain at least one lowercase, UPPERCASE, digit, and symbols (@#\-_$%^&+=§!.) Minimum length of 8 characters.

1. A login attempt of more than five trials, will result in access denial of the account from the user.
2. A change of password without correctly getting the previous password will result in access denial of the account from the user.
3. Changing of user's password periodically else the account will be blocked.

Password Hashing and One Time Password

For the proposed research, the user's password and DES key will be hashed utilizing Secure Hash Algorithm 256 (SHA-256) and put away while the Key-Hash Message Authentication Code – Message Digest 5 (HMAC-MD5) will be used to perform a One-Time Password Challenge-Response authentication mechanism utilizing the user's password as key and the unique One Time Password as message for each authentication.

Procedure/Algorithm in achieving the Research Goals

This procedure is divided into three stages whichincludes registration, authentication and password recovery. The procedure concludes that no malicious action was conducted given a straight forward path.

1. **Registration** INPUT: Sign UpBegin

a. Input a valid username, phone number and password(password that follows the password policies given)

b. Compare username, phone number and passwordinputted.

c. Compute DES key by hashing the user's password

.

d. Set timer for next change of password, thus change of DES key.

e. Return login user and redirects to dashboard.End.

OUTPUT: Login user and redirect to dashboard

2. **Authentication** INPUT: User loginBegin

a. Input a valid username and password.

b. Compare username sent first to server

c. Compute row with username and a random one-timepassphrase and stores in table on Server

d. Return passphrase back to client

e. Compute passphrase with password

f. Return passphrase with newly hashed password toserver

g. Compare newly hashed password by hashingpassword in the table on the Server with passphrase

h. If login attempt is more than 5,

i. If username and password is invalid, lock account fora period of time.

j. Else If username and password are valid

k. Return user is logged in.End.

OUTPUT: User is logged in

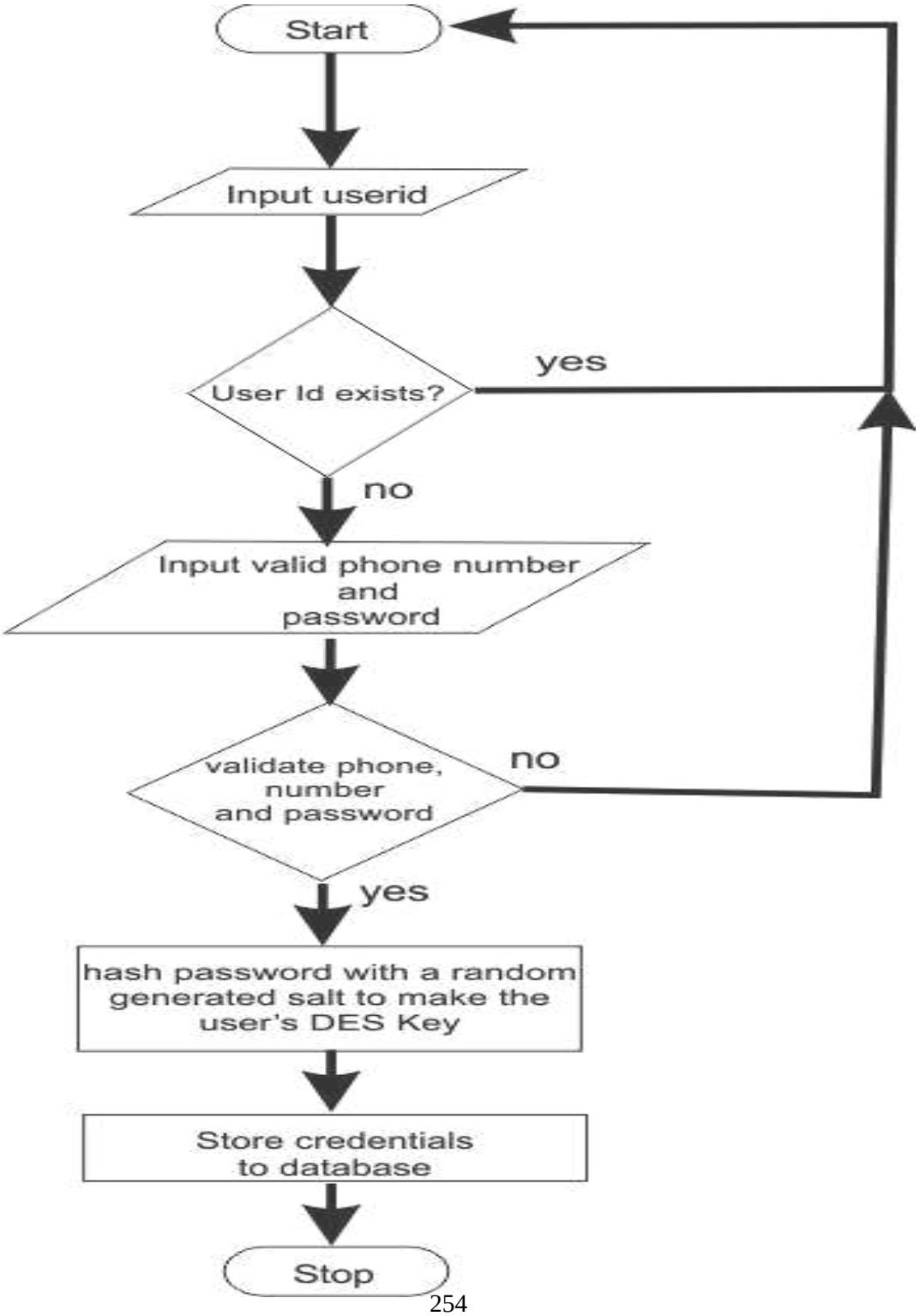
3. **Password Recovery** INPUT: valid usernameBegin

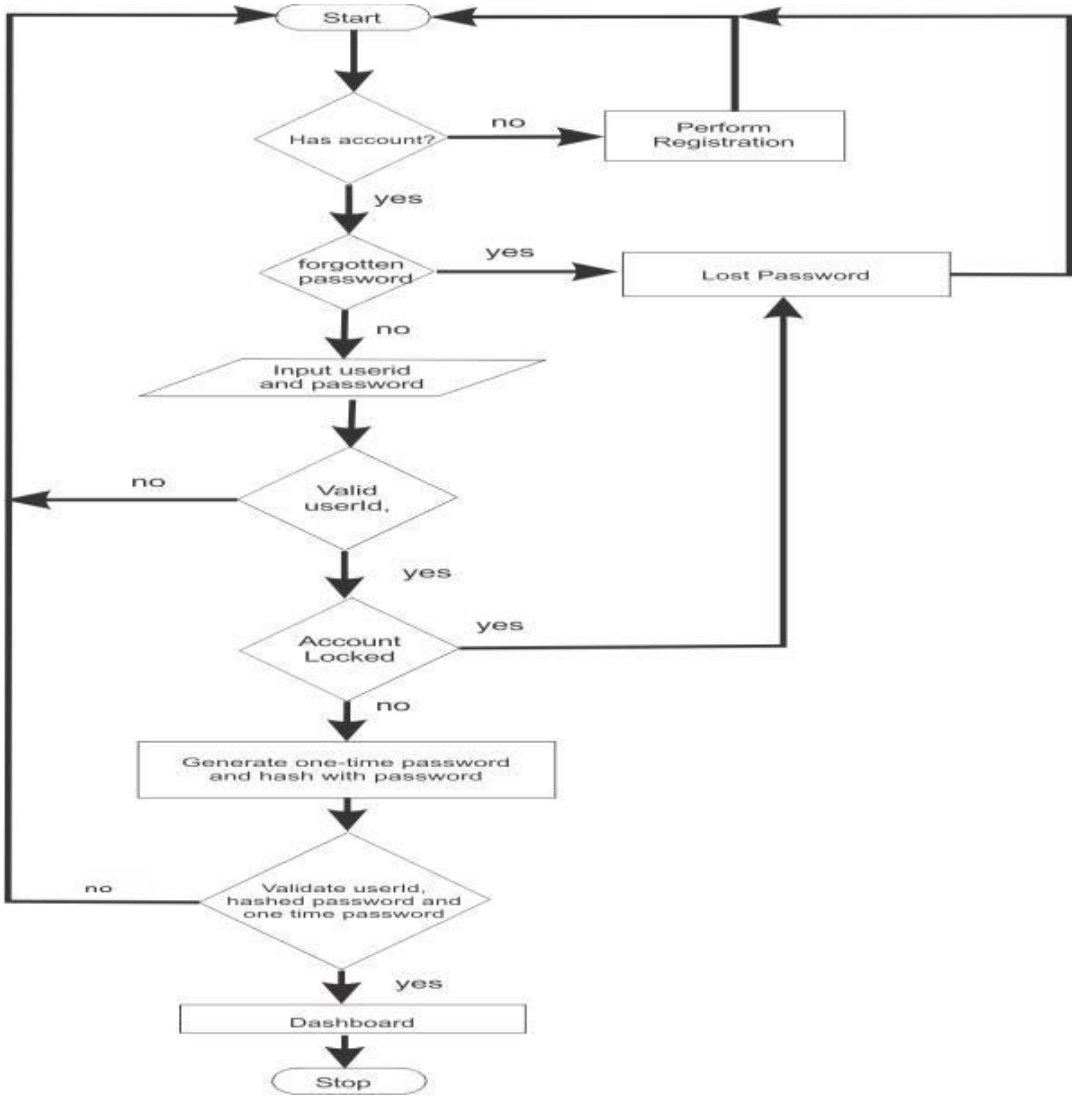
a. Input valid username

b. Notification is sent to user email,

c. Account is unlocked.End.

OUTPUT: account is unlocked


Start
Input userid
User Id exists?
yes
no
Input valid phone number
and
password
validate phone,
number
and password
no
yes
hash password with a random
generated salt to make the
user's DES Key
Store credentials
to database
Stop


Start
Has account?
no
Perform Registration
yes
forgotten password
yes
Lost Password
no
Input userid and password
Valid userId,
no
yes
Account Locked
yes
no
Generate one-time password and hash with password
Validate userId, hashed password and one time password
no
yes
Dashboard
Stop

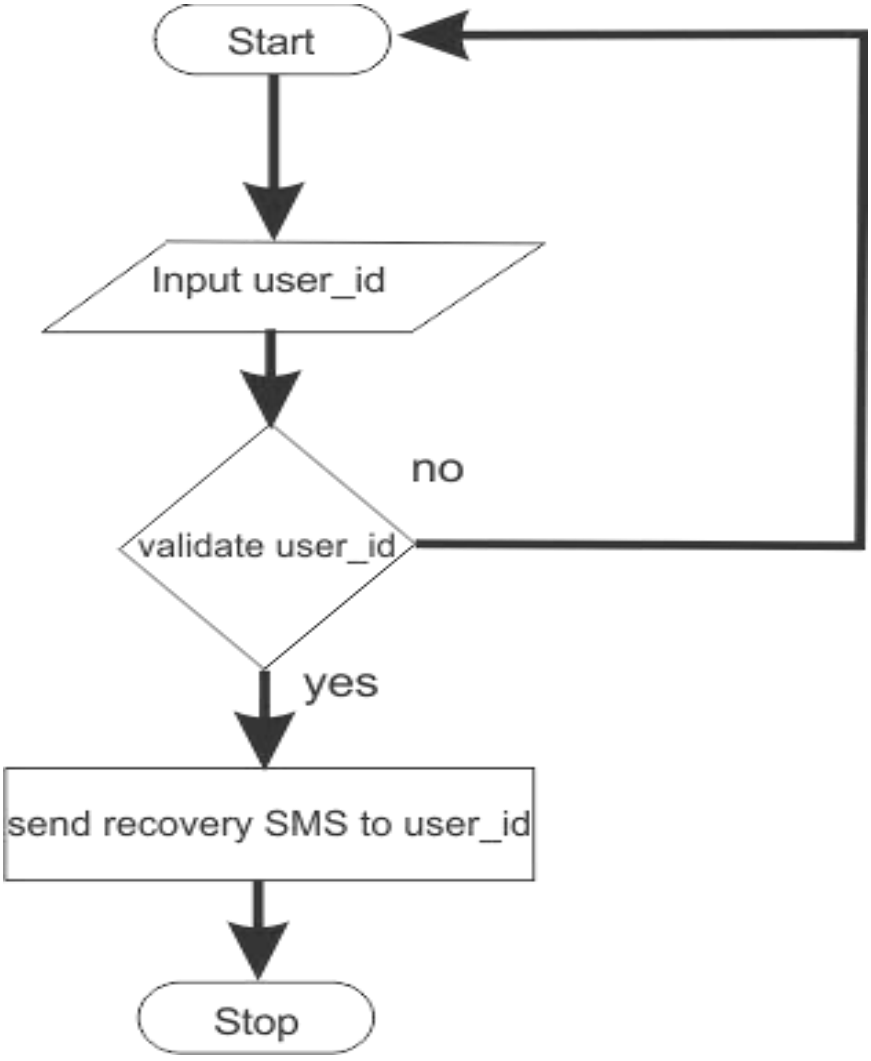


Fig: 4.16 Authentication using OTP and Password Recovery

Register Credentials

By entering the required information—a valid username, which is an email address that is not already in use by another user, a valid phone number for sending SMS messages to users, and a password that strictly complies with the password policy set forth by the system—users are able to create accounts into the system. There is a link in this interface that can take users to the login page.

Access the System

Users can enter their information into a form on this interface to be authenticated by the system. Following the implementation of the password policy, the user accounts will be temporarily disabled after 5 failed tries. This page has a link that directs users to the registration page.

View User's Credential

In this the information displayed includes: the username, phone, date password was updated, latest date to change password, count down timer to when to change date and the user's unique Data Encryption Standard Key.

Edit User's Credential

The user can edit the fundamental information using this interface. Username, phone, and password are among the information that can be changed. The DES key for that user will change when the password is changed, and both the password updated and next update fields will always be updated. A user's account will be blocked if they enter the incorrect password five times after updating their password.

CHAPTER 5: FINDINGS AND CONCLUSIONS

Once the proposed NGVPN model has been successfully implemented in a framework, various findings and interpretations have been identified. Traffic was created by the framework and the model has been identified as working properly and capable of resolving all the desired results. Themodel was implemented in different places to offer the best solutions to the traditional VPN. Theconsumer has obtained satisfactory results.

5.1 Findings from Identity Management Analysis –

Concatenate Media Access Control(MAC) address to IPv6 and testing with various protocols

Survey Results: Cent OS Linux, Mac OS, Ubuntu Linux, Windows OS, and Windows Server systems have been used for surveys. In the following sections, the results of each of the tests are described.

5.1.1 Host System Information

CentOS: The Cent OS IPv6 feature has been tested on a Windows OS host virtual machine fromthe VirtualBox. This test system was used to install the OS "factory default" by a Cent OS mirrorprovided by RIT without system updates or installation of any additional software. The VM network interface was directly connected to the host wired Ethernet port and all of the traffic entering the host could be monitored. By default on system network interfaces, IPv6 functionalitywas enabled. The system's Ethernet MAC Address was 08:00:27:f5:00:87 for the local area connection.

Mac OS: A physical Apple MacBook Pro 3,1 laptop running OS X 10.6 Build 10A432 was testedwith IPv6 functionality in Mac OS. Instead of system updates or additional software, it used the factory's default OS installation. The OS' IPv6 feature is supplied by a customized version of KAME's network stack, which has its IPv6 network stack as '20010528/apple-darwin' version identified. By default on system network interfaces, IPv6

functionality was enabled. The system'swired network port Ethernet MAC address was 00:1b:63:9a:1b:67.

Ubuntu OS: IPv6 feature was tested on a Windows XP Professional SP3 host on a VirtualBox virtual machine. Installing the OS provided by RIT's Ubuntu mirror "factory default" used the testing system without any system updates or additional software installed. The VM network interface was directly connected to the host wired Ethernet port and all of the traffic entering the host could be monitored. By default on system network interfaces, IPv6 functionality was enabled. The Ethernet MAC address was 08:00:27:a4:89:65 for the Local Area Interface of the system.

Microsoft Windows OS: The Windows OS functionality of IPv6 was tested on the virtual VirtualBox machine. This test system used the Microsoft Developer Network Academic Alliance(MSDNAA) factory default installation OS, without system updates, service packs, or the installation of further software. The VM network interface was directly connected to the host wiredEthernet port and all of the traffic entering the host could be monitored. By default on system network interfaces, IPv6 functionality was enabled. The system Local Area Interface Ethernet MAC address was 08:00:27:05:9a:35.

Microsoft Windows Server: IPv6 functionality has been tested on a Windows XP Professional SP3 host on a Virtual Box virtual machine. The test system used the fabric default MSDNAA OSinstallation without installing any system updates or additional programs. Microsoft Developer Network Academic Alliance (MSDNAA). The VM network interface was directly connected to the host wired Ethernet port and all of the traffic entering the host could be monitored. By defaulton system network interfaces, IPv6 functionality was enabled. The system's Local Area InterfaceEthernet MAC Address was 08:00:27:f0:ae:2c.

5.1.2 NDP Behavior Results

Generation of NDP Solicitations

1) The identified values of the IPv6 and ICMPv6 options and flags

a) Cent OS system sends NDP solicitation packets used NULL values for the IPv6 and Flow Labels fields, 255 for the IPv6 packet field Hop Limit, and null values for the ICMPv6 datagram code field. This is as per IETF RFC4861 guidelines.

b) Mac OS NDP Solicitation packets sent to the Mac OS system used null values for the IPv6 package traffic class and flow label, 255 for the IPv6 packet hop limit, and a null value for the ICMPv6 datagram code field. This is as per IETF RFC4861 guidelines.

c) The Ubuntu system solicitations used NULL values for the fields Traffic and Flow Label in theIPv6 packet, 255 for the Hop Limit field in the IPv6 packet, and the NULL value for the ICMPv6 datagram for the code field. This is as per IETF RFC4861 guidelines.

d) Windows OS: NDP Packets sent by Windows system use NULL values on IPv6 packet for Traffic Class and Flow Labels, 255 for IPv6 Hop Limit, and a null value on the ICMPv6 data graph for the fields Code and Flags. This is as per IETF RFC4861 guidelines.

e) The Windows Server solicitation packet used NULL values for the IPv6 packet traffic class andflow label, 255 for the Hop limit field for the Ipv6 packet, and a zero value for the ICMPv6 fields.This is as per IETF RFC4861 guidelines.

2)The IPv6 Address is used as the Source

a) Cent OS: The Cent OS solicitation packets used a Source IPv6 address in the same subnet as the target address, such as: 2001:db8:1111:222:: /64 subnet node solicitation packets, 2001:db8:1111:222:a00:27ff:fef5:87 Source address. On the other hand, the system's connection address is used as the Source IPv6 address for Multicast NDP Solicitations.

b) Mac OS: For example, when soliciting nodes on 2001:db8:1111:222:::/64 subnet, 2001:db8:1111: 2222::21b:63ff:fe9a:2454 was used as a source address. All the packets sent by Mac OS use the Source IPv6 address that existed in the same subnet as Target.

c) Ubuntu: Source address 2001::db8:1111:221:: /64 Subnet, 2001:db8:111:222:a00:27ff:fa4:8965was used as a Source address so that the Multicast NDP solicitation packages sent by the Ubuntu OS were a Source IPv6 address that was present on the same subnet as the Target address. On the other hand, the system's connection address is used as the Source IPv6 address for Unicast NDP Solicitations.

d) Windows OS: NDP Solicitation Packets for the Windows OS system were based on the Ipv6 address that was in the same subnet as the IPv6 Source address; for example, the 2001: db8:1111:222:/64 subnet node request was used as the Source Address for 2001: db8:1111:1222:9c9:47da:29a7:d834.

e) Windows Server: In order to request a node from 2001:DB8:1111:222::/64 subnet, 2001:DB8:1111:211:f9ca:6411:b670:7715 was used as a Source address. NDP Solicitation Packetfrom a window server uses an IPv6 address that is present on the same subnet as the Target as an IPv6 Source address.

3) The values used for IPv6 and Ethernet Destinations in multicast Solicitations are as follows:

a) Cent OS: Multicast NDP Solicitation packets sent by the Cent OS used the proper Solicited Node Multicast addresses in the Destination field for both the IPv6 and Ethernet headers, rather than the All Nodes Multicast IPv6 address (FF01::1) or broadcast Ethernet address (FF:FF:FF:FF:FF:FF). For example, a multicast Solicitation to fc00:aaaa:aaaa:2222:217:31ff:feb8:F21C used the IPv6 Destination address of ff02::1:ffb8:F21C and Ethernet Destination address of 33:33:ff:b8:F21C.

b) Mac OS: Multicast NDP Solicitation packets sent by the Mac OS used the proper Solicited NodeMulticast addresses in the Destination field for both the IPv6 and Ethernet headers, rather than theAll Nodes Multicast IPv6 address (FF01::1) or broadcast Ethernet address (FF:FF:FF:FF:FF:FF). For example, a multicast Solicitation to fc00:aaaa:aaaa:2222:217:31ff:feb8:F21C used the IPv6 Destination address of ff02::1:ffb8:F21C and Ethernet Destination address of 33:33:ff:b8:F21C.

c) Ubuntu: Multicast NDP Solicitation packets sent by the Ubuntu used the proper Solicited NodeMulticast addresses in the Destination field for both the IPv6 and Ethernet headers, rather than theAll Nodes Multicast IPv6 address (FF01::1) or broadcast Ethernet address (FF:FF:FF:FF:FF:FF). For example, a multicast Solicitation to fc00:aaaa:aaaa:2222:217:31ff:feb8:F21C used the Destination IPv6 address of ff02::1:ffb8:F21C and Ethernet Destination address of 33:33:ff:b8:F21C.

d) Windows OS: Multicast NDP Solicitation packets sent by the Windows OS used the proper Solicited Node Multicast addresses in the Destination field for both the IPv6 and Ethernet headers,rather than the All Nodes Multicast IPv6 address (FF01::1) or broadcast Ethernet address (FF:FF:FF:FF:FF:FF). For example, a multicast Solicitation to fc00:aaaa:aaaa:2222:217:31ff:feb8:F21C used the IPv6 Destination address of ff02::1:ffb8:F21C and Ethernet Destination address of 33:33:ff:b8:F21C.

e) Windows Server: Multicast NDP Solicitation packets sent by the Windows Server used the proper Solicited Node Multicast addresses in the Destination field for both the IPv6 and Ethernet headers, rather than the All Nodes Multicast IPv6 address (FF01::1) or broadcast Ethernet address(FF:FF:FF:FF:FF:FF). For example, a multicast Solicitation to fc00:aaaa:aaaa:2222:217:31ff:feb8:F21C used the IPv6 Destination address of ff02::1:ffb8:F21C and Ethernet Destination address of 33:33:ff:b8:F21C.

4) Identify the Source Link-Layer Address option used in the ICMPv6 datagram are as follows:

a) Cent OS: The Cent OS has properly implemented the ICMPv6 Source link-layer address option as outlined by IETF RFC4861, for all unicast and multicast NDP solicitation packets.

b) Mac OS: The ICMPv6 Link-Layer Address option outlined by IETF RFC4861 was corrected to all NDP Unicast and Multicast Solicitation Packets sent to Mac OS.

c) Ubuntu: An ICMPv6 Source link-layer address, as outlined by IETF RFC4861, was properly deployed for all NDP Unicast and Multicast Solicitation packets sent by the Ubuntu OS.

d) Windows OS: The ICMPv6 Source link-layer address option was correctly implemented as outlined by IETF RFC4861 for all the packets sent by the Windows OS to solicited unicast and Multicast NDP.

e) Windows Server: The ICMPv6 Source link-layer address option, as outlined in IETF RFC4861,has correctly been implemented by all NDP unicast and multi-cast solicited packets sent by the Windows Server.

Generation of NDP Advertisements

1) The values of the IPv6 and ICMPv6 options and flags are as follows:

a) Cent OS: NDP advertising packets sent from Cent OS use a null value for the Ipv6 packet's Traffic Class and Flow Label fields, 255 for an IPv6 packet Hop Limit field, and null for an ICMPv6 datagram code field. When a unicast solicited was received, the advertisement was set with the solicited flag, and both the requested and the override flag were used in response to a multicast solicitation.

b) Mac OS: The Mac OS advertising packets for NDP use null for the IPv6 packet traffic class andflow label fields, 255 for the IPv6 Hop Limit field, and the ICMPv6 datagram's code field null value. The advertisement set the solicited flag to respond to a unicast solicitation and set both Solicited and Override flags to respond to a multi-cast solicitation.

c) Ubuntu: Ubuntu used the null value of the Traffic Class and Flow Label fields in the IPv6 packetNDP advertisement packets, the 255 value for the IPv6 packet Hop Limit fields, and the null valuefor the ICMPv6 data field. When a unicast solicitation was received, the advertisement was set with the requested flag, and both the requested and the override flag were used in response to a multicast Solicitation.

d) Windows OS: The null value of the Traffic Class and Flow Label fields for the IPv6 packet, 255 for the Hop Limit field for an IPv6 packet, and a code field for the ICMPV6 datagram used by the Windows OS system advertisement packets. The flags solicitation and overridden were setfor Unicast and Multicast solicitations in the ICMPv6 part.

e) Windows Server: The Windows Server NDP Advertising packets use null values in the IPv6 package for traffic and flow label fields; 255 for the IPv6 packet Hop Limit field; and a null valuefor the ICMPv6 datagram code field. The flags solicited and overridden were set for answers to the two in the ICMPv6 package.

Unicast and Multicast Solicitations.

2) IPv6 Address used in the following Source.

a) Cent OS: The target address of the solicitation packets is used as an IPv6 source address for allNDP advertisements (Unicast, Multicast) sent by Ubuntu.

b) Mac OS: The link-local interface address used in IPv6 Source for all Mac OS (unicast andmulticast) advertisements sent to Mac OS, not the requested destination.

c) Ubuntu. The Target address from the solicitation packet used by the Ubuntu as the IPv6 Sourceaddress for all NDP advertises (unicast and multicast).

d. Windows OS: The target address from the solicitation packet is utilized as the source address ofIPv6 by all NDP advertises (unicast and multicast) sent by a Windows OS.

e. Windows Server: The target address for a solicitation Packet is used as the IPv6 source addressin all NDP advertisements (unicast and multicast) sent from the Windows Server.

3) Target Link-Layer Address is the option used in the ICMPv6 datagram, details as follows:

a) Cent OS: the Cent OS included the ICMPv6 datagram with the Target link-layer address only when the NDP solicitation response is generated. All unicast NDP solicitations excluded all options for ICMPv6.

b) Mac OS: Mac OS included in the ICMPv6 datagram only the Target link-layer address option when creating a multi-cast NDP solicitation response. All unicast NDP solicitations excluded all options for ICMPv6.

(c) Ubuntu: The Ubuntu included in the ICMPv6 datagram a Target link-layer address option only when a multi-cast NDP solicitation was generated. All unicast NDP solicitations excluded all options for ICMPv6.

d) Windows OS: When generating multicast and unicast NDP solicitation reactions, the WindowsOS has included the Target link-layer address feature in the ICMPv6 datagram.

e) Windows Server: When generating a multicast and unicast NDP solicited response, the Windows Server included the Target link-layer address option in the ICMPv6 datagram.

Handling of Incoming NDP Solicitations

1. The system adds the Source address to its NDP table based on this Solicitation.

a) Cent OS: Yes, the NDP cache system with Stale State added to the Source IPv6 and Source Link-Layer addresses of the received NDP solicitation. This is as per IETF RFC4861 recommendation that an entry is created if a solicitation does not come from an unspecified addressand the Source Link-Layer Address option is available.

b) Mac OS: Okay, the NDP system cache with Stale State has been added to the system IPv6 Source Link-Layer addresses of the received NDP solicitation. This conduct is consistent with theIETF RFC4861 recommendation which states that if the request is not from an unspecified location and the source link-layer address option is present, it should be created. The following is not recommended.

c) Ubuntu: Yes, the NDP cache system with Stale status was added to the Source IPv6 and SourceLink-Layer addresses of the received NDP solicitation. This is as per IETF RFC4861 recommendation that an entry is created if a solicitation does not come from an unspecified addressand the Source Link-Layer Address option is available.

d) Windows 7: Yes, in the Stale state system NDP cache was added the Source IPv6 and the SourceLink-Layer addresses from the NDP Received solicitation. This is as per IETF RFC4861 recommendation that an entry is created if a solicitation does not come from an unspecified addressand the Source Link-Layer Address option is available.

e) Windows Server: Yes, the NDP cache system with Stale state has been added with the source IPv6 and source link-layer addresses for the received NDP solicitation. This is as per IETF RFC4861 recommendation that an entry is created if a solicitation does not come from an unspecified address and the Source Link-Layer Address option is available.

2. The target system is reciprocated with an NDP solicitation to the test system by this solicitation packet.

a) Cent OS: The Cent OS attempted to get the entry to an accessible state by sending its own NDPSolicitation packets to the Testing Workstation when the Stale entry was added to the NDP cache. Even when no other traffic has been sent to workstations, this reciprocal finding has happened.

b) Mac OS: The Mac OS tried to bring the entry to the Reachable state by sending its own unicastNDP solicitation packets to the testing workstation once the stale entry was added to the NDP cache. Even when no other traffic has been sent to workstations, this reciprocal finding has happened.

c) Ubuntu: Once the NDP cache has been added, the Ubuntu has tried by sending its own unicastNDP application packets to a testing workstation to bring the entry to a Reachable state. Even when no other traffic has been sent to workstations, this reciprocal finding has happened.

d) Windows OS: The Windows OS attempted to get the entry into the Reachable state by sendingthe NDP Unicast solicitation Packets to a testing workstation after Stale's entry was added to the NDP cache. Even when no other traffic has been sent to workstations, this reciprocal finding has happened.

e) Windows Server: The Windows Server has tried to bring a Reachable entry by sending its ownunicast NDP solicitation packet to the testing workstation when Stale entry is added to the NDP cache. Even when no other traffic has been sent to workstations, this reciprocal finding has happened.

3. The following system will respond to malformed Solicitations.

a) ICMPv6 Code field is not 0.

1. Cent OS: No, an Advertisement was not sent.

2. Mac OS: No, an Advertisement was not sent.

3. Ubuntu: No, an Advertisement was not sent.

4. Windows OS: No, an Advertisement was not sent.

5. Windows Server: No, an Advertisement was not sent

b) ICMPv6 Incorrect Checksum.

1. Cent OS: No, an Advertisement was not sent.

2. Mac OS: No, an Advertisement was not sent.

3. Ubuntu: No, an Advertisement was not sent.

4. Windows OS: No, an Advertisement was not sent.

5. Windows Server: No, an Advertisement was not sent.

c) ICMPv6 Source Link-Layer Address option missing.

1. Cent OS: Yes, unicast advertisements are sent, but only after the system has found the rightMAC address via its NDP solicitation.

2. Mac OS: Yes, a unicast ad has been sent, but only after the MAC address has been found throughthe system's NDP soliciting multicast.

3. Ubuntu: Yes, a unicast ad has been sent, but the proper MAC address has only been found bythe system through its NDP soliciting Multicast.

4. Windows OS: Yes, a unicast ad was sent, but only after a MAC address was found by its ownmulticast NDP solicitation.

5. Windows Server: Yes, unicast publicity has been sent but the correct MAC address has beenfound only after the system has received a multi-cast NDP solicitation.

d) ICMPv6 Source Link-Layer Address Option incorrect.

1. Cent OS: Yes, the Source Link-Layer Address value of the solicitation was sent a unicastadvertisement (rather than the Source MAC address).

2. Mac OS: Yes, the Source Link-Layer address value of the application was sent a singleadvertisement (rather than the Source MAC address).

3. Ubuntu: Yes, the Source Link-Layer Address value of the solicitation was sent a unicastadvertisement (rather than the Source MAC address).

4. Windows OS: Yes, a unicast ad was sent to the Source Link-Layer Address value of theapplication (rather than the Source MAC address).

5. Windows Server: Yes, the Source Link-Layer Address value of the solicitation was sent aunicast advisory (rather than the Source MAC address).

e) IPv6 Destination to All Nodes (not Solicited Node).

1. Cent OS: Yes, a unicast Advertisement was sent as normal.

2. Mac OS: Yes, a unicast Advertisement was sent as normal.

3. Ubuntu: Yes, a unicast Advertisement was sent as normal.

4. Windows OS: Yes, a unicast Advertisement was sent as normal.

5. Windows Server: Yes, a unicast Advertisement was sent as normal.

f) IPv6 Destination to incorrect Solicited Node.

1. Cent OS: No, an Advertisement was not sent.

2. Mac OS: No, an Advertisement was not sent.

3. Ubuntu: No, an Advertisement was not sent.

4. Windows OS: No, an Advertisement was not sent.

5. Windows Server: No, an Advertisement was not sent.

g) IPv6 Hop Limit field not 255.

1. Cent OS: No, an Advertisement was not sent.

2. Mac OS: No, an Advertisement was not sent.

3. Ubuntu: No, an Advertisement was not sent.

4. Windows OS: No, an Advertisement was not sent.

5. Windows Server: No, an Advertisement was not sent.

h) IPv6 Flow Label field not 0.

1. Cent OS: Yes, a unicast Advertisement was sent as normal, with a null Flow Label field.

2. Mac OS: Yes, a unicast Advertisement was sent as normal, with a null Flow Label field.

3. Ubuntu: Yes, a unicast Advertisement was sent as normal, with a null Flow Label field.

4. Windows OS: Yes, a unicast Advertisement was sent as normal, with a null Flow Label field.

5. Windows Server: Yes, a unicast Advertisement was sent as normal, with a null Flow Label field.

i) IPv6 Traffic Class field not 0.

1. Cent OS: Yes, a unicast Advertisement was sent as normal, with a null Traffic Class field. 2.Mac OS: Yes, a unicast Advertisement was sent as normal, with a null Traffic Class field.

3. Ubuntu: Yes, a unicast Advertisement was sent as normal, with a null Traffic Class field.

4. Windows OS: Yes, a unicast Advertisement was sent as normal, with a null Traffic Class field.

5. Windows Server: Yes, a unicast Advertisement was sent as normal, with a null Traffic Classfield.

j) Ethernet Destination to Broadcast.

1. Cent OS: Yes, a unicast Advertisement was sent as normal.

2. Mac OS: Yes, a unicast Advertisement was sent as normal.

3. Ubuntu: Yes, a unicast Advertisement was sent as normal.

4. Windows OS: Yes, a unicast Advertisement was sent as normal.

5. Windows Server: Yes, a unicast Advertisement was sent as normal.

k) Ethernet Destination to incorrect Solicited Node.

1. Cent OS: No, an Advertisement was not sent.

2. Mac OS: No, an Advertisement was not sent.

3. Ubuntu: No, an Advertisement was not sent.

4. Windows OS: No, an Advertisement was not sent.

5. Windows Server: No, an Advertisement was not sent.

Handling of Incoming NDP Advertisements

1. **The system accepts unsolicited NDP Advertisements when no cache entry exists.**

a) Cent OS: No, if the appropriate entry is not found in the NDP system cache, the Cent OS won'taccept an unexpected unrequested NDP advertisement packet.

b) Mac OS: Unknown, methods used to test this system have not properly generated unwanted adsas described in the IETF RFC4861.

c) Ubuntu: No, when a suitable input is not found in a system's NDP cache, the Ubuntu will notaccept an unexpected unwanted NDP advertisement packet.

d) Windows OS: No, if the appropriate entry is not found in the NDP cache of the system, theWindows OS will not accept an unforeseen NDP Advertisement packet.

e) Windows Server: No, if the appropriate system entry is unavailable in the NDP cache, theWindows Server System shall not accept an unexpected NDP Advertisement packet.

2. The system accepts malformed Advertisements. (Solicited, non-gratuitous)

a) ICMPv6 Code field, not 0.

1. Cent OS: No, the advertised MAC was not added to the system's NDP cache.

2. Mac OS: No, the advertised MAC was not added to the system's NDP cache.

3. Ubuntu: No, the advertised MAC was not added to the system's NDP cache.

4. Windows OS: No, the advertised MAC was not added to the system's NDP cache.

5. Windows Server: No, the advertised MAC was not added to the system's NDP cache.

b) ICMPv6 Incorrect Checksum.

1. Cent OS: No, the advertised MAC was not added to the system's NDP cache.

2. Mac OS: No, the advertised MAC was not added to the system's NDP cache.

3. Ubuntu: No, the advertised MAC was not added to the system's NDP cache.

4. Windows OS: No, the advertised MAC was not added to the system's NDP cache.

5. Windows Server: No, the advertised MAC was not added to the system's NDP cache.

c) ICMPv6 Target Link-Layer Address option missing.

1. Cent OS: No, the advertised MAC was not added to the system's NDP cache.

2. Mac OS: No, the advertised MAC was not added to the system's NDP cache.

3. Ubuntu: No, the advertised MAC was not added to the system's NDP cache.

4. Windows OS: No, the advertised MAC was not added to the system's NDP cache.

5. Windows Server: No, the advertised MAC was not added to the system's NDP cache.

d) ICMPv6 Target Link-Layer Address option incorrect.

1. Cent OS: Yes, value-added to the system NDP cache of the advertising's Target Link-Layeraddress (rather than the Source MAC address).

2. Mac OS: Yes, in the NDP cache of the system was added the advertisement's target link-layeraddress (rather than the Source MAC address).

3. Ubuntu: Yes, in the cache of the NDP system the value Target Link-Layer Address has beenadded (rather than the Source MAC address).

4. Windows OS: Yes, the Link-Layer Address value of the Advertisement was added to the NDPcache of the system (rather than the Source MAC address).

5. Windows Server: Yes, to the system NDP cache the Advertisement Target Link-Layer Addressvalue has been added (rather than the Source MAC address).

e) ICMPv6 Flag field combinations (Router, Solicited, and Override).

1. Router only:

a) Cent OS: The OS Cent accepted the NDP Advertisement packet to send the ping request with aTarget Link-Layer Address value. However, in the NDP cache with the INCOMPLETE State, theIPv6 and MAC addresses are added, forcing an immediate set of NDP solicitations.

b) Mac OS: Yes, the system accepts the NDP Adversity Package and uses the ping application value of the Target Link-Layer address. However, the NDP cache of the system has not been checked so it is unknown if the entry has been saved.

(c) Ubuntu: Ubuntu accepted the NDP advertising packet for sending ping requests using a TargetLink-Layer address. However, in the NDP cache with the INCOMPLETE State, the IPv6 and MAC addresses are added, forcing an immediate set of NDP solicitations.

d) Windows OS: The Windows OS accepted the NDP advertising packet and was using the ping request value from the Target Link-Layer Address. However, in the NDP cache with the INCOMPLETE State, the IPv6 and MAC addresses are added, forcing an immediate set of NDP solicitations.

e) Windows Server: The Windows server system accepted the NDP Advertisement packet and used the ping application target Link Address value. However, in the NDP cache with the INCOMPLETE State, the IPv6 and MAC addresses are added, forcing an immediate set of NDP solicitations.

2. Override only:

a) Cent OS: The OS Cent accepted the NDP Advertisement packet to send the ping request with aTarget Link-Layer Address value. However, in the NDP cache with the INCOMPLETE State, theIPv6 and MAC addresses are added, forcing an immediate set of NDP Solicitation.

b) Mac OS: Yes, the system accepts the NDP Adversity Package and uses the ping request value of the Target Link-Layer Address. However, the NDP cache of the system has not been checked so it is unknown if the entry has been saved.

(c) Ubuntu: Ubuntu accepted the NDP advertising packet for sending ping requests using a TargetLink-Layer address. However, in the NDP cache with the INCOMPLETE State, the IPv6 and MAC addresses are added, forcing an immediate set of NDP Solicitations.

d) Windows OS: The Windows OS has accepted the NDP advertising packet and used the ping request value for the target link-layer address. However, in the NDP cache with the INCOMPLETE State, the IPv6 and MAC addresses are added, forcing an immediate set of NDP Solicitations.

e) Windows Server: The Windows Server has accepted a packet NDP advertisement and has usedthe ping request value from the Target Link-Layer Address. However, in the NDP cache with theINCOMPLETE State, the IPv6 and MAC addresses are added, forcing an immediate set of NDP Solicitations.

3. Not solicited (none):

a) Cent OS: The OS Cent accepted the NDP Advertisement packet to send the ping request with aTarget Link-Layer Address value. However, in the NDP cache with the INCOMPLETE State, theIPv6 and MAC addresses are added, forcing an immediate set of NDP solicitations.

b) Mac OS: Yes, the system accepts the NDP Adversity Package and uses the ping request value of the Target Link-Layer Address. However, the NDP cache of the system has not been checked so it is unknown if the entry has been saved.

(c) Ubuntu: Ubuntu accepted the NDP advertising packet for sending ping requests using a TargetLink-Layer address. However, in the NDP cache with the INCOMPLETE State, the IPv6 and MAC addresses are added, forcing an immediate set of NDP Solicitations.

d) Windows OS: The Windows OS has accepted the NDP advertising packet and used the ping request value for the target link-layer address. However, in the NDP cache with

the INCOMPLETE State, the IPv6 and MAC addresses are added, forcing an immediate set of NDP solicitations.

e) Windows Server: The Windows Server has accepted a packet NDP advertisement and has usedthe ping request value from the Target Link-Layer Address. However, in the NDP cache with theINCOMPLETE State, the IPv6 and MAC addresses are added, forcing an immediate set of NDP Solicitations.

4. Override and Solicited:

a). Cent OS: Yes, the IPv6 and MAC addresses of the NDP cache have been accepted and have been added in the Rechargeable state.

b) Mac OS: Yes, the system accepts the NDP Adversity Package and uses the ping request value of the Target Link-Layer Address. However, the NDP cache of the system has not been checked so it is unknown if the entry has been saved.

c) Ubuntu: Yes, NDP advertising has been accepted and IPv6 and MAC Address with Reachablestate have been added to NDP cache.

d) Windows OS: Yes, NDP has been accepted advertisement and the IPv6 and MAC addresses have been added to the Reachable NDP cache.

e) Windows Server: Yes, IPv6 and MAC addresses were added to an accessible state of the NDP cache.

f) IPv6 Multicast to All Nodes.

1. Cent OS: No, the advertised MAC was not added to the system's NDP cache.

2. Mac OS: No, the advertised MAC was not added to the system's NDP cache.

3. Ubuntu: No, the advertised MAC was not added to the system's NDP cache.

4. Windows OS: No, the advertised MAC was not added to the system's NDP cache.

5. Windows Server: No, the advertised MAC was not added to the system's NDP cache.

g) IPv6 Multicast to Solicited Node Address.

1. Cent OS: No, the advertised MAC was not added to the system's NDP cache.

2. Mac OS: No, the advertised MAC was not added to the system's NDP cache.

3. Ubuntu: No, the advertised MAC was not added to the system's NDP cache.

4. Windows OS: No, the advertised MAC was not added to the system's NDP cache.

5. Windows Server: No, the advertised MAC was not added to the system's NDP cache.

h) IPv6 Source not matching ICMPv6 Target.

1. Cent OS: Yes, the value of the advertising target address in the NDP cache with a reachablestate was added to the system (rather than the Source IPv6 address).

2. Mac OS: Yes, in a system NDP cache the advertising's target address value with a ReachableState was added (rather than the Source IPv6 address).

3. Ubuntu: Yes, the value of the target address advertisement has been added to the system'sReachable NDP cache (rather than the Source IPv6 address).

4. Windows OS: Yes, the Advertisement Target Address value has been added to the ReachableState of System NDP cache (rather than the Source IPv6 address).

5. Windows Server: yes, a target address value of the advertisement has been added to the system'sReachable state NDP cache (rather than the Source IPv6 address).

i) IPv6 Hop Limit field not 255.

1. Cent OS: No, the advertised MAC was not added to the system's NDP cache.

2. Mac OS: No, the advertised MAC was not added to the system's NDP cache.

3. Ubuntu: No, the advertised MAC was not added to the system's NDP cache.

4. Windows OS: No, the advertised MAC was not added to the system's NDP cache.

5. Windows Server: No, the advertised MAC was not added to the system's NDP cache.

j) IPv6 Flow Label field not 0.

1. Cent OS: Yes, to a system NDP cache with an accessible state, the IPv6 and MAC addresseshave been added.

2. Mac OS: Yes, the system NDP cache with Reachable state has added the IPv6 and MACaddresses.

3. Ubuntu. Yes, in the system's Reachable NDP cache, IPv6 and MAC addresses have been added.

4. Windows OS: Yes, the system's Reachable NDP cache has added both IPv6 and MAC addresses.

5. Windows Server: Yes, the system NDP cache with Reachable state was supplemented with IPv6and MAC addresses.

k) IPv6 Traffic Class field not 0.

1. Cent OS: Yes, to a system NDP cache with an accessible state, the IPv6 and MAC addresseshave been added.

2. Mac OS: Yes, the system NDP cache with Reachable state has added the IPv6 and MACaddresses.

3. Ubuntu. Yes, in the system's Reachable NDP cache, IPv6 and MAC addresses have been added.

4. Windows OS: Yes, in a system NDP cache with a Reachable status the IPv6 and MAC addresseshave been added.

5. Windows Server: Yes, the system NDP cache with Reachable state was supplemented with IPv6and MAC addresses.

5.1.3 Multiple IPv6 Addresses Results

1. For every prefix in the router advertisement, the system will generate the address plus anaddress with the link-local scope.

a) Cent OS: Yes, for every subnet prefix in the router advertisement packets, this system generateda stateless auto-configuration address.

b) Mac OS: Yes, for each subnet prefix in router advertising packets, the system generated a stateless auto-configuration address.

c) Ubuntu: Yes, for each subnet prefix contained in the Router Advertisement packets, the systemgenerated a Stateless Autoconfiguration address.

d) Windows OS: Yes, for each of the subnet prefixes in the Router Advertisement packet the system has generated an IPv6 address. However, these were not Stateless Address Auto settings based on the Ethernet MAC address of the interface. In place of the MAC address in the SAA configuration process, the system generates a separate randomized interface identifier.

e) Windows Server: Yes, for each of the prefixes included in the advertisement packets, the systemgenerated an IPv6 address. However, these were not Stateless Address Auto settings based on theEthernet MAC address of the interface. In place of the MAC address

in the SAA configuration process, the system generates a separate randomized interface identifier.

2. When pinging nodes on the same network segment, the Source addresses are used as follows:

(a) Cent OS: When sending traffic to the local network segment node, the host system always employed an address on the same subnet as the destination, e.g. when sending the node to 2001:db8:1111:222::1, its 2001:db8:111:222::/64 subnet used its address. Both the NDP solicitation Packet and the Ping Request Package sent to each destination were using the same source address.

b) MAC OS: When sending traffic to the local network sector node, the host system always uses an address on the same subnet as the intended destination; e.g., when sending it to 2001:db8:1111:222::1, in the subnet 2001:db8:111:222::/64, its address used. Both the NDP solicitation Packet and the Ping Request Package sent to each destination were using the same source address.

c) Ubuntu: The host system always used an address for traffic to a local network node in the samesubnet as the desired destination; e.g., it used its 2001:db8:1111:2222::1 subnet address for sendingto host::db8:1111:222::/64 subnet address. Both the NDP solicitation Packet and the Ping RequestPackage sent to each destination were using the same source address.

d) Windows OS: During my observations, when transmitted traffic to the local network segment,the host system usually uses an address within the same subnet as the destination; e.g. when it was transmitted to:1111.11:222::1, the host system used its 2001:db8:1111:222::/64 subnet address. However, the unique-local address of the system was used for the NDP solicitation source in a fewcases when a link-local address was ping (this behavior could not be reliably reproduced). In all cases, an address in the same subnet as the Destination address was used by the ping request packet.

e) The Windows Server: When sending traffic to the local net bound node, the host system alwaysused the same subnet address as the intended destination; e.g. when

sending it in 2001:db8:111:2222::1, its 2001:db8:111:222::/64 subnet address was used. Both the NDP solicited Packet and the Ping Request Package sent to each destination were using the same sourceaddress.

3. When pinging nodes on a different network segment, the Source addresses used are as follows:

a) Cent OS: The Cent OS host is used as the Source IPv6 address for all transmits to destinations beyond its network, except for those starting with FC00 and its Global-scope address (2001:db8:1111:222,a00:27ff:fef 5:87). As the source IPv6, its Unique-Local address was used for destinations beginning with FC00 (fc00:aaaa:aaaa:222:a00:27ff:fef5:87). For NDP solicited packets to the gateway and the Ping Request Packs to the destination, the same address was used for each of these cases.

b) Mac OS: When transmitting traffic to a node, not on the local network segment the host systemappeared to use a non-link-local address that was numerically closest to the designated destinationaddress as the source address. For example, the address of the host in fc00:aaa::aaa::2222::/64 wasused as the Source when sending Ping Requests to 9999::1 (not active on the network and no trafficreply). In 2001:db8:1111:222:::/64, however, host addresses were used as the Source by sending Ping requests to 0099::1 (again, not active in the network). The routable address nearest to the destination address has been used in both examples (2001:: is closer to 0099:: than FC00::, and FC00:: is closer to 9999:: than 2001::). For the NDP solicitation packet at the gateway as well as the ping request packet at the target, the same Source address was used.

c) Ubuntu: The Ubuntu Host has used its Global-scope address for all traffic outside its local network, except for the traffic that begins with its FC00, (2001:db8:1111:222:a00:27ff:fea4:8965),as its IPv6 source address. The Source IPv6 address was used as a Unique-local address for destinations commencing with the FC00 (fk00:aaaaa:aaaa:222:a00:27ff:fea4:8965). For NDP solicitation packets to the gateway

and the Ping Request Packs to the destination, the same addresswas used for each of these cases.

d) Windows OS: When sending traffic to a node, not on the local network segment, the host systemseemed to be using a non-link-local address that was numerically closest to that intended destination. The FC00:AAAA:AAA:222:::/64 subnet was used as the Source, for example, when sending Ping Requests to 9999::1 (not active in the network, therefore no traffic reply). However,the host address of 2001:DB8:111:222:::/64 was used as the Source when Ping requests were sentto 0099::1 (again not active on the network). The routable address nearest to the destination addresshas been used in both examples (2001:: is closer to 0099:: than FC00::, and FC00:: is closer to 9999:: than 2001::). In some cases, however, the system used the link-local address for the NDP gateway request. At all the other times, both the NDP Gateway Request Packet and the Ping Request packet used the same source address.

e) Windows Server: When sending traffic to a node, not on the local network segment a host system seemed to be using the non-link-local address numerically close to the target destination address as a source address. The FC00:AAAA:AAA:222:::/64 subnet was used as the Source, for example, when sending Ping Requests to 9999::1 (not active in the network, therefore no traffic reply). However, the host address of 2001:DB8:111:222:::/64 was used as the Source when Ping requests were sent to 0099::1 (again not active on the network). The routable address nearest to the destination address has been used in both examples (2001:: is closer to 0099:: than FC00::, andFC00:: is closer to 9999:: than 2001::). For the NDP solicitation packet at the gateway as well as the ping request packet at the target, the same Source address was used.

5.1.4 Privacy Extensions Results

1. Privacy Extensions for IPv6 have supported

a) Cent OS: Yes, Privacy Extensions are supported by the CentOS.

b) Mac OS: Yes, Privacy Extensions are supported by the Mac OS10.6 system.

c) Ubuntu: Yes, Privacy Extensions are supported by Ubuntu.

d) Windows OS: Yes, Privacy Extensions are supported by the Windows OS.

e) Windows Server: Yes, Privacy Extensions are supported by the Windows Server.

2. Privacy Extensions for IPv6 are enabled by default:

a) Cent OS: No, it does not default option to activate the Data Extensions functionality. Rather, the hosts generate only a single IPv6 address by the observed network prefix as per the Stateless Address Auto setup guidelines.

b) Mac OS: No, it does not default option to enable Privacy Extensions. Rather, the hosts generateonly a single IPv6 address by the observed network prefix as per the Stateless Address Auto setupguidelines.

c) Ubuntu: No, the functionality of Privacy Extensions is not activated by default. Rather, the hostsgenerate only a single IPv6 address by the observed network prefix as per the Stateless Address Auto setup guidelines.

d) Windows OS: Yes, Windows OS systems have privacy extensions enabled by default. Temporary addresses for each of the network prefixes in the observed router advertising will be created, in addition to the stateless address automatic configuration addresses.

e) Server for Windows No, the functionality for privacy extensions is not enabled by default. Rather, only one IPv6 address is generated for hosting by a respected network prefix in line with the Stateless Address Autoconfiguration instructions.

3. The procedure is used to activate/deactivate the use of Privacy Extensions for IPv6 are asfollows:

a) Cent OS: Cent OS allows for a multi-step process to enable Privacy Extensions.

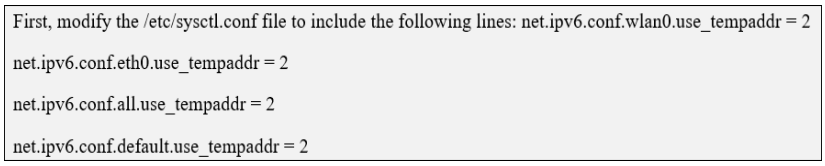


Figure 5.1: Changes to enable Privacy Extensions

Alternatively, run the following commands in the terminal:

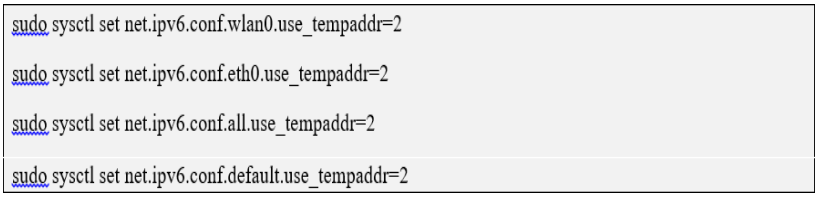


Figure 5.2: Changes to enable Privacy Extensions via command

Use the Stateless Address Autoconfiguration Address as shown in Figure 5.2 After a restart, each networking interface automatically generates a pseudo recent temporary address. Likewise, the Privacy Extensions feature can be disabled by removing /etc/sysctl.conf lines, issuing "=0" ratherthan "=2" commands and restarting the system. The OS does not provide graphical utilities for changing confidentiality extensions.

b) Mac OS: To enable privacy extensions on Mac OS using a **sudo sysctl -w net.inet6.ip6.use tempaddr=1** command in the terminal.

Once this command is issued, the network interfaces must be restarted by rebooting the system orissuing the terminal commands **sudo ifconfig** and **sudo ifconfig up**. After resetting the network, in addition to the stateless address Autoconfiguration, the interface automatically creates a pseudoalternative temporary address. Privacy extensions can also be deactivated if the **sudo sysctl -w net.inet6.ip6.use tempaddr=0** is issued Terminal command and network resetting. The OS doesnot provide graphical utilities for changing confidentiality extensions.

c) Ubuntu: Enabling Ubuntu Privacy Extensions is a multi-stage process.

To include the following lines, change the “/etc/sysctl.conf” file first:

```
net.ipv6.conf.wlan0.use_tempaddr = 2
net.ipv6.conf.eth0.use_tempaddr = 2
net.ipv6.conf.all.use_tempaddr = 2
net.ipv6.conf.default.use_tempaddr = 2
```

Figure 5.3: Enabling Ubuntu Privacy Extensions

Alternatively, execute the following commands in the terminal:

```
sudo sysctl set net.ipv6.conf.wlan0.use_tempaddr=2
sudo sysctl set net.ipv6.conf.eth0.use_tempaddr=2
sudo sysctl set net.ipv6.conf.all.use_tempaddr=2
sudo sysctl set net.ipv6.conf.default.use_tempaddr=2
```

Figure 5.4: Enabling Ubuntu Privacy Extensions

Finally, in addition to stateless address self-configuration, each network interface automatically generates a temporary pseudorandom address. After a reboot. Likewise, the Privacy Extensions feature can be disabled by removing /etc/sysctl.conf lines, issuing "=0" rather than "=2" commandsand restarting the system. The OS does not provide graphical utilities for changing confidentialityextensions.

d) Windows OS: The Windows OS privacy extension function can be deactivated by the **netsh interface command ipv6 set privacy state=disabled** prompt. The network interfaces should deactivate the use of existing temporary addresses automatically after reboot the system. Similarly,Privacy Extensions can be re-activated through a command prompt with the **netsh interface ipv6set privacy state=enabled**. The OS does not provide graphical utilities for changing confidentiality extensions.

e) Windows Server: Privacy Extensions can be activated by emitting the **netsh interface command ipv6 set privacy state=activated** by prompt commands. Once the system reboots, existing temporary addresses for all known network prefixes are generated automatically by network interfaces. In the same way, Privacy Extensions can be disabled by issuing the command prompt with the **netsh interface ipv6 set Privacy State=disabled**. The OS does not provide graphical utilities for changing confidentiality extensions.

4. The default parameters relating to Privacy Extensions are as follows:

CentOS: The system will use a Valid Lifetime of seven days and a Preferred Lifetime of one day for the temporary addresses it generates once the Privacy extensions functionality is enabled. Thisis as per IETF RFC 4861 recommendations.

b) Mac OS: The system uses a maximum of seven-day valid life and a maximum of one-day preferred lifetime for each temporary address it produces when privacy extensions are activated. This is as per IETF RFC4861 recommendations.

c) Ubuntu: The system shall use a maximum seven-day valid lifetime and a maximum Preferred Lifetime of one day for each temporary address that it generates when confidentiality extensions functionality is enabled. This is as per IETF RFC 4861 recommendations.

d) Windows OS: The system will use a maximum seven-day Valid Lifetime and a maximum one-day Life Preferred time for each temporary address that is created when privacy extension functionality is activated. This is as per IETF RFC4861 recommendations.

e) Windows Server: The system will use a maximum seven-day valid lifetime and a maximum preferred one-day lifetime for each generated time address when the privacy extensions functionality is enabled. This is as per IETF RFC4861 recommendations.

5. The Parameters that can be customized are as follows:

a) Cent OS: Besides the privacy extension enabling and disabled feature, sysctl allows to alter theValid and Preferred Lifetimes of the temporary addresses generated. Figure 5.5 shows sysctl parameters for IPv6 privacy extensions.

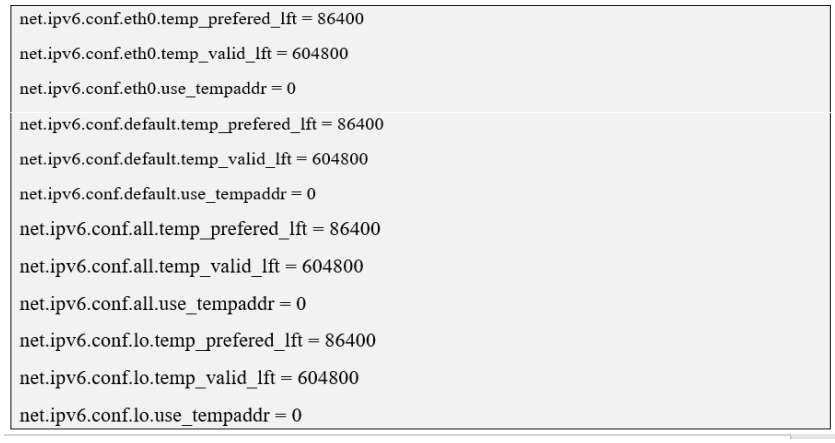


Figure 5.5: sysctl parameters for IPv6 privacy extensions for IPv6

b) Mac OS: Aside from the privacy extension feature enabled and disabled, the sysctl utility lets change the Valid and Preferred Lifetime of the temporary addresses created. Figure 5.6 shows thesysctl parameters for IPv6, with bold-type privacy extensions.

Figure 5.6: Mac OS X's sysctl parameters relating to Privacy Extensions for IPv6

c) Ubuntu: Apart from the privacy extensions enabled and disabled functionality, the sysctl utilityenables to change the valid and preferred lives used in temporary addresses generated. Figure 5.7displays the sysctl parameters for IPv6 privacy extensions.

```
net.ipv6.conf.all.use_tempaddr = 2
net.ipv6.conf.all.temp_valid_lft = 604800
net.ipv6.conf.all.temp_prefered_lft = 86400
net.ipv6.conf.default.use_tempaddr = 2
net.ipv6.conf.default.temp_valid_lft = 604800
net.ipv6.conf.default.temp_prefered_lft = 86400
net.ipv6.conf.lo.use_tempaddr = 2
net.ipv6.conf.lo.temp_valid_lft = 604800
net.ipv6.conf.lo.temp_prefered_lft = 86400
net.ipv6.conf.eth0.use_tempaddr = 2
net.ipv6.conf.eth0.temp_valid_lft = 604800
net.ipv6.conf.eth0.temp_prefered_lft = 86400
```

Figure 5.7: Ubuntu sysctl parameters relating to Privacy Extensions for IPv6

d) Windows OS: Apart from activating and deactivating Privacy Extensions, the netsh utility letschange the maximum valid and preference lifetimes and numbers of tentative duplicate address detection attempt to the time, random time, and max time regeneration parameters. Figure 5.8 shows the netsh parameters for Privacy Extensions for IPv6.

```
C:\Users\rama>netsh interface ipv6 show privacy
Querying active state...
Temporary Address Parameters
----------------------------------------------
Use Temporary Addresses                 : enabled
Duplicate Address Detection Attempts    : 5
Maximum Valid Lifetime                  : 7d
Maximum Preferred Lifetime              : 1d
Regenerate Time                         : 5s
Maximum Random Time                     : 10m
Random Time                             : 0s
```

Figure 5.8: Windows OS netsh parameters relating to Privacy Extensions for IPv6

e) The Windows Server: is a tool that allows to alter the maximum Valid and Primary Data Life times and number of Duplicate Address Detection attempts attempted for time-consuming, random time and max random time parameters for temporary address regeneration in addition to allowing for disabling and enables privacy extensions. e) Figure 5.9 shows the Netsh parameters for IPv6 privacy extensions.

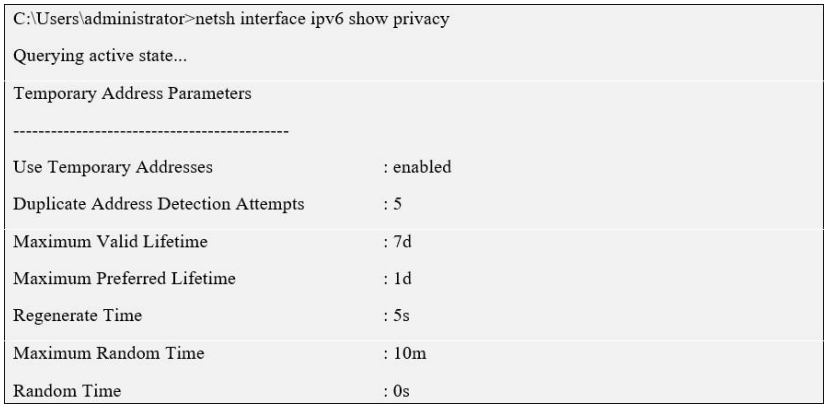


Figure 5.9: Windows Server netsh parameters relating to Privacy Extensions for IPv6

6. The Temporary addresses created for each network prefix and scope

a) Cent OS: The Cent OS generates temporary addresses for all known network prefixes exceptthe prefix of link-local scope.

b) Mac OS: For all known network prefixes the Mac OS will create temporary addresses exceptfor the link-local scope prefix.

c) Ubuntu: for all known network prefixes except the prefix for link-local scope, Ubuntu willgenerate temporary addresses.

d.) Windows OS: A temporary address for all known network presets except the link-local scopeprefix is generated in the Windows OS system.

e) Windows Server: The Windows Server system generates temporary addresses, except the link-local scope prefix, for all known network prefixes

7. The Temporary addresses used for all outbound traffic on all scopes are as follows:

A) Cent OS: Yes, all host outbound traffic and connections will use the newest time address as the Source Address for the corresponding network prefix when privacy extensions functionality is activated.

b) Mac OS: Yes. All outbound traffic and host connections using the latest temporary address for their respective network prefix, when privacy extensions functionality is enabled, as a Source address.

Ubuntu: Yes, all the outbound traffic and links from the host used the latest temporary address for the corresponding network prefix for the source address when the Privacy Extensions functionalityhas been activated.

d) Windows OS: Yes, all outbound traffic and connections that originate from the host will use, with the appropriate network prefix, the newest temporary address as the source address when Privacy Extensions functionality is enabled.

e) Windows Server: Yeah, all outgoing traffic and connections from the host will use the newest time address for the corresponding network prefix as the Source address when Privacy Extensionsfunctionality is activated.

5.2 Implementation of DHCPv6 Findings

DHCPv6 is default with support for the MAC address of RFC 3315. The python code is written tobind Mac with DHCPv6.

5.2.1 Basic Usage

NOTE: This was tested on Windows WSL and Ubuntu 18.04. Other platforms may or may not work.

DORA is a fundamental abbreviation for the DHCP methodology of operations.

Discover Message / ssDHCPDISCOVER
Offer Message / DHCPOFFER
Request Message/ DHCPREQUEST
Acknowledgment Message/ DHCPACK

Figure 5.10: DORA is a fundamental abbreviation

Port 67 requires the clients to configure the IP on clients using UDP on the server and port 68. Currently, another program is bound to check whether port 68 (and port 67 if the relay is set) is (for example with: sudo netstat -tulpn). If something is connected to these ports, they must be killed first and avoided restarting.

To see all available options, run dora.py with a flag -h/—help:

```
$ sudo dora.py -h
usage: dora.py [-h] [-i INTERFACE] [-a MAC_ADDR] [-d] [-u] [-s SERVER] [-r RELAY] [-v] [-o OPTIONS] [-p PORT] [--
target_port TARGET_PORT] [-@ TARGET]

optional arguments:
  -h, --help            show this help message and exit
  -i INTERFACE, --interface INTERFACE
                        Interface to bind to and make DHCP requests
  -a MAC_ADDR, --mac_addr MAC_ADDR
                        MAC address to use (default random)
  -d, --debug           Print debug statements
  -u, --unicast         Send DHCP packets over unicast to specified server
  -s SERVER, --server SERVER
                        Server to send DHCP packets. Required for unicast and for relay use.
  -r RELAY, --relay RELAY
                        Address to set the giaddr field to
  -v, --verbose         Verbosity level (v: show ack packet, vv: show all packets, vvv: show debug)
  -o OPTIONS, --options OPTIONS
                        JSON body of options to include in requests
  -p PORT, --port PORT  Port to send packets from on client machine
  --target_port TARGET_PORT
                        Port to send to on target machine
  -@ TARGET             Given an IP address of a DHCP server, sends unicast requests
```

Figure 5.11: dora options

NOTE: dora.py must be able to bind to port 68 (and 67 under certain circumstances) to function properly. This may require the use of sudo. This may also require stopping any services (e.g., systemd-networkd) that are already bound to those ports.

Limitation: the DHCP RFC 2131 sets the client port to 68 and the server port to 67 options that set different client or server ports are not expected to work with an RFC-compliant server.

```
rama@server:~$ ifconfig
eth1      Link encap:Ethernet  HWaddr 00:0e:09:86:99:96
          inet addr:192.168.1.9  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::6403:25cf:5c97:5d44/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:516872 errors:0 dropped:0 overruns:0 frame:0
          TX packets:319652 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:215973829 (215.9 MB)  TX bytes:68594936 (68.5 MB)
```

Figure 5.12: Ethernet configuration details

dhcpy6d solves this problem by tracking the link addresses and their corresponding MAC addresses when receiving a request by reading the IPv6 neighbor cache.

If the Link-Local Address and its MAC are yet to be found, dhcpy6d will attempt to obtain this information from the nearest cache. The command /usr/sbin/ndp -a -n is executed with Linux. Thislooks like the neighbor cache:

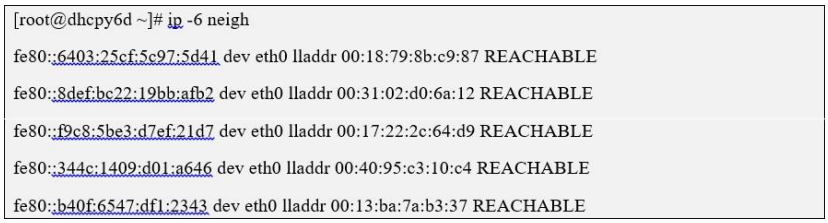


Figure 5.13: IPv6 Neighbor details

Finally, the DHCPv6 Server has been configured to provide IPv6 embedded with MAC address toall the DHCP clients configured in the LAN.

5.3 Implementation of VPN with IPv6 Configuration

For the entire framework configuration, the implementation of VPN is required. The entire UbuntuOS model was configured and open source utilities are used. The following points must be taken into account:

1. It was decided to openvpn with version 2.3 for implementation of VPN since ipv6 issupported.
2. The 'easy-rsa' utilities have opted for SSL/TLS certificates to implement with the publickey infrastructure.
3. The openvpn set to IPv4 is initially used for testing the correct configuration of the VPNserver and then the PKI and IPv6 integrations are carried out further.
4. DHCPv6 and OpenVPN synchronization are necessary for the VPN client generation ofIPv6.
5. On port 1194, the VPN server was set up.
6. Single port for remote client communication only.

The client would receive the IPv6 configuration based on DHCPv6 or the IP allocated to the VPNserver after IPV6 Implementation.

VPN implemented with IPv6 configuration finally successfully.

5.4 Customization of Packets using SDN

Another process for customizing all types of network packages is the software-defined network. The "OpenFlow" application for the implementation of the SDN part for the customization of IPv6packets is required. In addition, mininet (OpenFlow Switch) and OpenDaylight are required.

OpenFlow: OpenFlow is an open standard for a communication protocol that enables the controlplane to break off and interact to increase functionality and programmability with the forwardingplane of several devices.

OpenFlow Connection Sequence

- The switch may start a connection to the IP and default port of the controller, or a user-specified port (TCP 6633 pre-OpenFlow 1.3.2, TP 6653 post).
- The connection requests can also be initiated by the controller, but this is not common.
- Established TCP or TLS connection
- Send the two to an OFPT HELLO with a version field populated
- The negotiated version to be used is calculated for both
- A message of OFPT ERROR is sent if this cannot be decided.
- • The controller sends OFPT FEATURES-REQUEST to collect the switch's Datapath ID,along with the switch functions for each of the supported versions.

That gives some idea of OpenFlow's fundamental operations.

5.5 Customization of IPv6 Packets

- This section customizes packets for IPv6. The Jumbogram utility is necessary for this.
- Let us build a packet, and send a spoofed source address to IPv6-Jumbogram.
- Python helps build a spoofed network packet.
- Results:
- The spoofing address is required • The packet transfer destination address is required forthe network • Place it in a spawning version of the standard Python variables •
- Create a header extension. The Jumbogram option must be placed in the Hop-By-Hopheader:
- Stack the headers, check and forward the result to the

sending featureUse IPv6 to send spoofed packets to the destination

via the network.

5.6 Implementation of End-to-End Encryption

The framework is also important to receive, transmit, and other network users and internally employees are unable to access the data under a highly secure channel. Due to confidentiality, information security practices are now the most important concern. The End-to-End Encryption implementation is one of the best methods of mitigating human attacks in the center.

In addition to other major python packages:

- Urllib
- cryptography

Findings:

The python code is a self-explanatory concept and the data transmission is encrypted between thesource and the destination.

The man-in-the-middle attack was mitigated according to the information security practices standards.

5.7 Two Factor Authentication

This is one of the other security features that the framework requires. Two Factor: 2FA Authentication is necessary when the customer logs first or an existing customer attempts to log onto any new device not listed in the database.

To implement and implement the 2FA python

language.Implementation of major libraries is:

- Secrets
- Hashlib
- Hmac
- Math
- Time

Findings:

The python code is self-explanatory for implementing the concept and for synchronizing the clientto each existing user with the database.

This automation model handles all devices and reports all possible activities in the log server.

5.8 HTTPS (SSL) packets on VPN-IPSEC

Under the google update, all websites are required to be ranked well via the https (SSL/TLS) protocol. The problem now is to identify the client-side digital certificate verification when the request for a URL goes to https, so that if there is a missing public

password on the browser, the digital certificate that is to be configured/saved on the client's browser is forwarded to the webserver. The certificate now delivers to the middle-end due to the VPN or any proxy services and should not be customer-configured. The configuration of VPN must therefore be integrated into the SSL/TLS support, to synchronize the whole Certificate to the VPN server with its clientsrequested. To mitigate IP spoofing and other network layer attacks, the framework should also besynchronized to VPN-IPSec.

For IKEv2 VPN Server implementation on Ubuntu 18.04 with StrongSwan

- Setup StrongSwan, a daemon should be set up as a VPN server, in open-source IPSec.
- Creating a Certificate Authority
- Generate key
- Generating a Certificate for the VPN Server
- Configuring VPN Authentication
- Finally restart the server

Finally, successfully VPN-IPSec with SSL implemented on the server-side.

5.9 Log Analysis

Log Analysis is the last and most important function of the framework. This feature includes all activities logged by the log server and a few master training algorithms used to train engineers to identify different attacks or common behavior of the attacker/botnets/clients. It allows to connectand perform different types of data.

Findings:

Lots of features implemented which are as follows:

1. Identifies system service problems with python system logs.

2. Further analysis of the web server logs and Mysql database by Python and Apache Spark.
    a. Weblog data analysis
    b. Analysis of HTTP status code
    c. c. Frequent hosts analysis c.
    d. Show the top 20 most common endpoints
    e. Show the top 10 endpoints of error.
    f. Total single hosts number
    g. Number of single hosts per day
    h. Number of single hosts per day Average number of requests per host per day
    i. 404 answer codes Counting
    j. The top 20 404 reply code endpoints are listed.
    k. The top 20 404 reply code is listed
    l. 404 errors per day are visualized
    m. Top 3 days for 404 mistakes

5.10 Top 5 VPNs analysis up to Nov 2021

1. Express VPN –With amazing security, lightning-fast bandwidth, and a wonderful user experience, this VPN company is the best overall. There were over 3000 servers available. AES-256 bit encryption, a private DNS, and a trusted server network The simultaneous support of up to 5 connections is possible. It costs just $6.67 a month and has a 30-day money-back guarantee (Save 49 percent and get 3 more months FREE with a 12-month plan).

2. Surfshark– The most cost-effective VPN is Surfshark, which offers a choice of inexpensive subscriptions. It features military-grade AES-256-Bit encryption, Multi-Hop,

Clean Web, and over 3000 servers spread across 94 different nations. Supporting countless connections concurrently for just $2.21/mo (Holiday Sale: 83 percent Off Plus

3. NordVPN– The most secure VPN on the market, with a variety of security features. It hasover 5100 servers in 60 countries, as well as features like Double VPN, CyberSec, Onion Over VPN, AES-256-Bit encryption, and safe DNS. Supports 6 simultaneous connections on a premium account for only $3.29/mo (Save 72 percent on a 2-year plan during the holiday season) with a 30-day money-back guarantee.

4. CyberGhost– VPN that is easy to use and has different settings for different purposes. It has over 7000 servers in 90 countries, AES-256-Bit encryption, and specialized streaming, torrenting, and online gaming modes. It costs $2.17/mo and comes with a 45-day money-backguarantee. It has 7 simultaneous connections.

5. PrivateVPN– A VPN provider with an emphasis on performance is perfect for a high-performing service. Along with 2048-bit encryption, a kill switch, port forwarding, IPv6 leak protection, application guard, obfuscation, and other capabilities, it boasts more than 200 servers spread over 68 nations. You may receive 6 simultaneous connections for for $2.07/month with a 30-day money-back guarantee.

5.11 Top 5 VPNs Comparison (Detailed Analysis – Nov. 2021)

We tested a number of VPNs, so we decided to compare them using the same standards as above to determine which one performed best.

VPN	Logs	Encryption	protocol	speed	Price	Trustpilot

Express VPN	No user logs & no data leaks	AES-256-Bit	OpenVPN, IKEv2, L2TP, Light way	ultra-fast 135 Mbps bandwidth	$6.67/month	9.3
Surf shark	Highest level of encryption, RAM-only servers, a strict no-logs policy,	AES-256-Bit	Wire Guard, IKEv2, and OpenVPN	FAST 58.46 Mbps	$2.21/month	9.0
Nord VPN	Well-protected servers and excellent firewall	AES-256-Bit	IKEv2/IPsec and OpenVPN protocols,	Average Speed 135MBPS	$3.71/month.	9.5
Cyber Ghost	Hides IP address and reroutes your internet traffic through An encrypted VPN tunnel.	AES-256-Bit	OpenVPN, L2TP/IPsec, IKEv2/ IPsec, and PPTP.	Higher than 90 Mbps	$12.99/per month	9.5
Private VPN	Strong VPN encryption between our VPN servers and your device.	AES-256-Bit	OpenVPN with UDP/TCP, L2TP, IPsec, PPTP	Higher than 90 Mbps	$3.50 /month	8.9

NGVPN	strong encryption between our VPN servers and device,. Easy to use ,and provides fast speeds for streaming	AES-256-Bit ML/AI Algorithm, End to Encryption through IPV6	OpenVPN, L2TP/ IPSec, IKEv2 / IPSec, and PPTP	ultra-fast 135 Mbps bandwidth	OPEN SOURCE	9.8

Table 5.1 : Comparison of VPN

Trustpilot value =Average Review

= Review from 2 companies is

= 9.9/10 and 9.8 /10

= 9.8/10

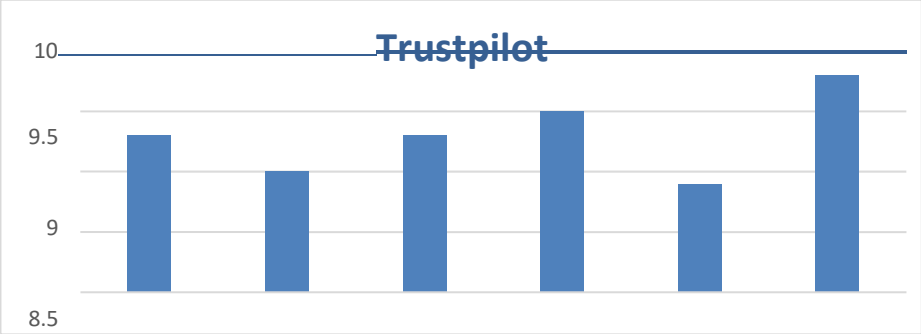


Figure 5.14: Comparison according to Trustpilot

Comparison criteria from following parameters:

The following is a list of the parameters we used to compare the VPN services in the spreadsheet we previously discussed **.**

• **Speed:** All VPN services are first rated as "Fast," "Average," or "Slow" on our VPN comparison chart. Look at the VPN speed test comparison table provided in the VPN company descriptions to determine which VPN service is the best.

• **Online Security and Privacy:** Virtual private networks (VPNs) are used to maintain internet security and privacy. Here are some of the factors we took into account while assessing VPNs for security and privacy.

• Jurisdiction: A VPN is obligated to abide by the laws of the nation in where it is located. The headquarters of a VPN are often in this nation. For instance, a VPN must maintain logs and abide by local laws if its corporate offices are in India.

• Privacy Statement:

To determine what kind of user data each VPN keeps, we carefully examined their privacy policies. The VPN you choose must not log any user data at all. Actually, the majority of the premium VPN services on this list merely keep track of the data required to manage the account and invoicing, including an email address. The top VPN comparison chart places VPNs that save timestamps, traffic logs, IP addresses, and other information lower. In our VPN comparison table below, you can find more details about these companies.

VPN	Surfshark	Express VPN	NORD VPN	PRIVATE VPN	VyprVPN	NGVPN
Average Speed Mbps	65.81	75.34	70.34	79.54	65.78	135

Jurisdiction	The British Virgin Islands	The British Virgin Islands	Panama	Sweden	Switzerland	India
Encryption	256-bit AES	256-bit AES	256-bit AES	256-bitAES	256-bit AES	256-bit AES
Simultaneous Connections	Unlimited	5	6	10	5	Unlimited
Money-Back Guarantee	30 days	30 days	45 days	30 days	30 days	Free

Table 5.2 : Compariso of VPN according to Speed , Jurisdiction, Encryption, Connection

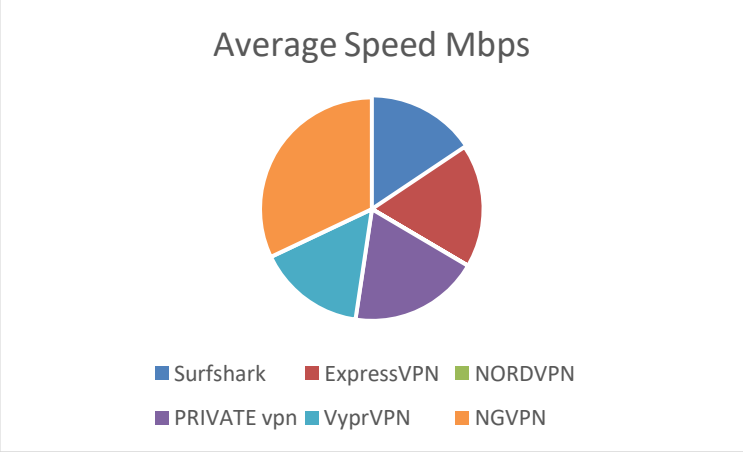


Figure 5.15 Comparison according to Speed

Comparison of VPN according to Overall Review:

Review	Express VPN	Surfshark	Nord VPN	Cyber ghost	Private VPN	NGVPN	

Excellent	85%	65%	71%	84%	85%	98%
Great	7%	17%	9%	7%	4%	4%
Average	2%	7%	4%	2%	1%	1%
Poor	1%	4%	4%	1%	1%	1%
Bad	5%	7%	2%	6%	3%	0%

Table 5.3 : Comparison of VPN according to Overall Review

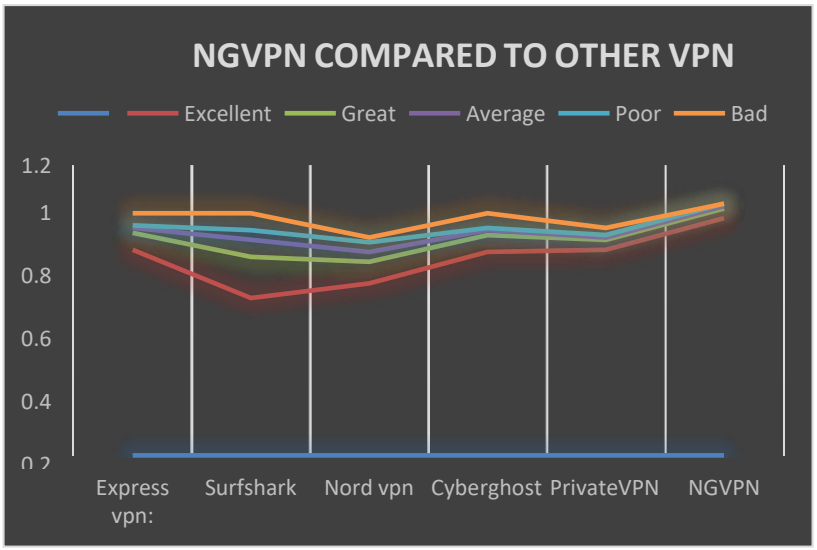


Figure 5.16 Comparison of NGVPN to other VPN

VPN IPv6 Solutions:

You basically have three alternatives when it comes to IPv6 and your VPN:

1. Use a VPN that supports IPv6.

2. Turn off IPv6 on your device.

3. Make use of a VPN that efficiently blocks IPv6 traffic.

The ideal approach is to choose a VPN provider that has a VPN server network that supports both IPv4 and IPv6. As we've already mentioned, only a few VPNs provide this service. Whenyour device connects to a VPN server with full IPv6 functionality, your device will broadcastboth an IPv4 and an IPv6 address**.**

The second option is to simply turn off IPv6 on your computer or device. For certain operatingsystems, but not all, this is a simple repair. On Windows, Mac OS, and Linux, for example, you can easily disable IPv6. However, many mobile devices today use IPv6 exclusively instead of IPv4. Similarly, "smart" devices that link to your network are vulnerable. of course, the third option is to use a VPN that effectively blocks IPv6 within the VPN client.

VPN Services:

VPN services are mostly dependent on whatever country you are in, as each has varying levelsof threat.

Every Internet user should take the next step and sign up for a VPN service. There is a compelling need to break free from the belief that our internet activities do not pose a harm tous.

The following are a few of the points that were processed in order to complete the Framework for Next-Generation VPN:

1. A Summary of Relevant Technologies
2. A review of the literature and a technical analysis of VPN, IPv6, Botnets, and other relevantand essential technologies.
3. Gathering information and resources for finalization and implementation.
4. Deployment of the foundational framework and infrastructure.

Comparison of VPN services

Provider	Countries	Servers	Technology	DNS	IPv6-leak	DNS hijacking
Hide My Ass	62	641	OpenVPN, PPTP	OpenDNS	Y	Y
IPVanish	51	135	OpenVPN	Private	Y	Y
Astrill	49	163	Open VPN, L2TP, PPTP	Private	Y	N
ExpressVPN	45	71	OpenVPN, L2TP, PPTP	Google DNS, Choopa Geo DNS	Y	Y
Strong	19	354	OpenVPN, PPTP	Private	Y	Y
PureVPN	18	131	OpenVPN, L2TP, PPTP	OpenDNS, Google DNS, Others	Y	Y
TorGuard	17	19	OpenVPN	Google DNS	N	Y
AirVPN	15	58	OpenVPN	Private	Y	Y
Private Internet Access	10	18	OpenVPN, L2TP, PPTP	Choopa Geo DNS	N	Y
VyprVPN	8	42	OpenVPN, L2TP, PPTP	Private (VyprDNS)	N	Y
Tunnelbear	8	8	OpenVPN	Google DNS	Y	Y
ProXPN	4	20	OpenVPN, PPTP	Google DNS	Y	Y
Mullvad	4	16	OpenVPN	Private	N	Y
Hotspot Shield Elite	3	10	OpenVPN	Google DNS	Y	Y
NGVPN	1	1	OpenVPN	Private	N	N

Table 5.4 : Comparison of VPN services

Many VPNs do not support the new version of IP protocol but rather block it. It is hard to havea great dual stack due to technical challenges. The best so far is **NGVPN** that works with IPv6.

Findings:

COVID PANDAMIC SURVEY RESULT

Based on the survey, initially collected all the details about commonly exposed vulnerabilities – CVE and categorized annually. It represents the present scenario ofthe impact generated by the attacks on VPN securities.

Year wise details	CVE's in Total	Top 5 CVE
020 – till (24th Aug 2020)	28	CVE-2020-6760, CVE-2020-5893, CVE-2020-
		5739, CVE-2020-5548, CVE-2020-5180
2019	53	CVE-2019-9955,CVE-2019-9657, CVE-2019-9584, CVE-2019-9461, CVE-2019-8459
2018	60	CVE-2018-9438,CVE-2018-9129, CVE-2018-8929, CVE-2018-8739, CVE-2018-7716

2017	26	CVE-2017-7935,CVE-2017-7738, CVE-2017-7344, CVE-2017-6784, CVE-2017-6620
2016	12	CVE-2016-6466,CVE-2016-6436, CVE-2016- 4945 CVE-2016-3887, CVE-2016-3657

Table 5.5 : Year wise exposed CVE's in total

In Total VPN – CVE's according to the NVD: 479.

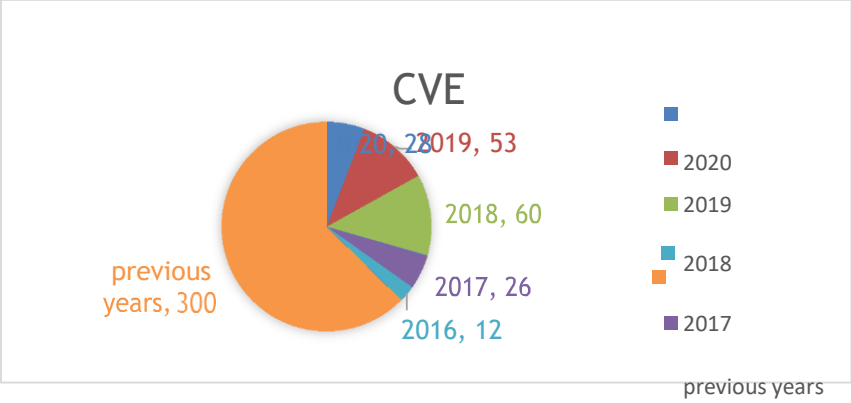


Figure 5.17 Year wise CVE- for VPN

Another analytical result is based on the common issues exposed to the existing Infrastructures in the year 2020 to date.

Identified Artifacts are as follows:

Issue Id.	Identified Common Issues in	Impact identified in- 2020	Total Exposed Vulnerabilities as per Issues
ID1	Software Versions	Allow remote attackers to cause a denial of service/reset Devices	1
ID2	Software Versions	Client responds to authentication requests over HTTP/ Unauthorized Access/ arbitrary File Deletion/ Effects to Client OS/ Remote execution	5
ID3	Hardware Firmware	Authenticated to remote command execution/ code execution/ Web based code execution/ arbitrary Code execution/ Authentication Bypass/ Failure	10
ID4	Hardware Models/ Services	Allow remote attackers to cause a denial of service/reset Devices/ Restart Devices/ Unstable/Reload/ Memory overflow / System Crashes/	6
ID5	Hardware basedServices	Allow an authenticated, local attacker to overwrite VPN profiles/ disconnect legitimate IPsec VPN sessions/ Digital Signature bypass/ Access protected Resources/ Session Hijacking/ LFI	6
	Total CVE:	In the year 2020 – till 24/ Aug./ 2020	28

Table 5.5 : Categorize Identified Issues with Impact on exposed CV in the year2020

S.no	CVE Issue ID	Issues Identified in
1	CVE-2020-6760	Schmid ZI 620 V400 VPN 090 routers ID 3
2	CVE-2020-5893	BIG-IP Edge Client, versions 7.1.5-7.1.8 ID2

3	CVE-2020-5739	Grandstream GXP1600 series firmware 1.0.4.152	ID 3
4	CVE-2020-5548	Yamaha LTE VoIP Router	ID4
5	CVE-2020-5180	Viscosity 1.8.2 on Windows and macOS	ID 2
6	CVE-2020-3435	(IPC) channel of Cisco AnyConnect Secure Mobility Client forWindows	ID5
7	CVE-2020-3398	(BGP) Multicast VPN (MVPN) implementation of Cisco NX-OSSoftware	ID4
8	CVE-2020-3397	(BGP) Multicast VPN (MVPN) implementation of CiscoNX-OS Software	ID1
9	CVE-2020-3358	(SSL) VPN feature for Cisco Small Business RV VPRouters	ID 4
10	CVE-2020-3357	(SSL) VPN feature of Cisco Small Business Routers	ID3
11	CVE-2020-3331	Web-based management interface of Cisco Wireless VPNRouters	ID 3
12	CVE-2020-3330	Cisco Small Business, VPN Firewall Routers	ID3
13	CVE-2020-3310	XML parser code of Cisco Firepower Device Manager	ID 4
14	CVE-2020-3220	Hardware crypto driver of Cisco IOS XE Software	ID5
15	CVE-2020-3189	VPN System Logging functionality for Cisco Firepower ThreatDefense (FTD) Software	ID4
16	CVE-2020-3146	Web-based management interface of the Cisco VPN Firewall	ID3
17	CVE-2020-3145	Web-based management interface of the Cisco VPNFirewall	ID3
18	CVE-2020-3144	Web-based management interface of the Cisco VPN Firewall	ID3
19	CVE-2020-3125	Kerberos feature of Cisco AdaptiveSecurity authentication Appliance (ASA) Software	ID4
20	CVE-2020-25043	Kaspersky VPN Secure Connection prior to 5.0	ID2
21	CVE-2020-2021	improper verification of signatures in PAN-OS SAML	ID 5
22	CVE-2020-2005	(XSS) vulnerability exists when visiting malicious websites withthe Palo Alto - VPN	ID5
23	CVE-2020-1987	Vulnerability in the logging component of Palo Alto	ID 5
24	CVE-2020-1631	Vulnerability in the HTTP/HTTPS service used by J-Web	ID5
25	CVE-2020-15467	Cohesive Networks vns3:vpn appliances before version4.11.1	ID 3
26	CVE-2020-13417	Privilege issue was discovered in Aviatrix VPN Client before 2.10.7	ID2
27	CVE-2020-12828	AnchorFree VPN SDK before 1.3.3.218	ID 2

28	CVE-2020-12812	Authentication vulnerability in SSL VPN in FortiOS 6.4.0, 6.2.0 to6.2.3, 6.0.9	ID3

Table 5.6 : List of CVE, categorized with Issue ID in the Year 2020

Based on the Analysis, It has been identified that the Firmware of mostly used VPN Hardware is vulnerable at a high scale.

Other listed Vulnerabilities also placed more impact on the VPN Routers/ Firewall/other Hardware's.

Mitigation Policies

N

According to the analytical results, the proposed mitigation policies are as follows:

- Update VPN hardware's with the latest Firmware's
- Update VPN hardware's with the latest versions of required services and SecurityPatches.
- Block all those services which are not in use.
- Maintain the proper logs for analysis purposes.
- Update software versions for client access.
- In case the client has the older version of any software connecting to the VPN, then in such a scenario either block the client or notify them for contingencies.
- Implement VPN infrastructure on IPv6 if possible instead of IPv4
- Strongly recommendation to apply 2 step Verification model for User Authentication.
- Monitor Network Bandwidth, Network packet transmission, with the connectionstatus.
- Analyze client's behavior based on various parameters like Duration,Accessed Hosts, Flow of data transmissions, IP Geo Location, etc.
- Implement updated secured tunnels and their protocols only for data transmissions.
- Maintain sessions with TLS encryption.

Limitations:

- IPv6 communication between members of various versions is not synchronized in the Multi-Version Cluster (MVC).
- LEA, ELA, CVP, UFP, SAM for these OPSEC Protocols it is not supported.
- ISP Redundancy is **not** supported if Dynamic Routing is configured

5.12 Findings

Lots of features implemented which are as follows:

1. Identifies system service problems with python system logs.

2. Further analysis of the web server logs and Mysql database by Python and ApacheSpark.

a. Weblog data analysis

b. Analysis of HTTP status code

c. Frequent hosts analysis

d. Show the top 20 most common endpoints

e. Show the top 10 endpoints of error.

f. Total single hosts number

g. Number of single hosts per day

h. Number of single hosts per day Average number of requests per host per day

i. 404 answer codes Counting

j. The top 20 404 reply code endpoints are listed.

k. The top 20 404 reply code is listed

l. 404 errors per day are visualized

m. Top 3 days for 404 mistakes

5.13 SUGGESTION:

IPv6 has larger addresses mostly and there is a significant security difference between IPv6 and IPv4. In some instances, IPv6 is slightly safer for link-local addresses. Security techniques and devices exist to implement and should be used to implement an IPv6 security policy.

Free VPNs have many problems, including slow connections, low reliability, and bandwidth limitations. One of the most important problems is that users can form part of a botnet without even being aware of it. This happened with the free VPN “Hola”, which was found to sell the bandwidth of its users to all types of shady organizations and to make them part of a botnet.

Its recommended to stay away from free VPNs and to prevent contingencies in the organizations.Instead, use a paid VPN as trustworthy as one also recommended. With these paid VPNs, the userwill be safe and secure without ever becoming part of a botnet.

IPv6 has larger addresses mostly and there is a significant security difference between IPv6 and IPv4. In some instances, IPv6 is slightly safer for link-local addresses. Security techniques and devices exist to implement and should be used to implement an IPv6 security policy.

Free VPNs have many problems, including slow connections, low reliability, and bandwidth limitations. One of the most important problems is that users can form part of a botnet without even being aware of it. This happened with the free VPN “Hola”, which was found to sell the bandwidth of its users to all types of shady organizations and to make them part of a botnet.

Its recommended to stay away from free VPNs and to prevent contingencies in the organizations.Instead, use a paid VPN as trustworthy as one also recommended. With these paid VPNs, the userwill be safe and secure without ever becoming part of a botnet.

5.14 CONCLUSION:

In this Framework, lots of policies are implemented for the Next Gen. VPN infrastructure. Further,the raw information is logged in the server and classified at the various stages in the form of datasetdetection, analysis, sanitization, purification, and further extends to dataset collection in a trainedengine. The next stage in compliance with the defensive handlers like firewalls, IDS, and IPS, etc.to mitigate the untraced threats in the VPN infrastructure. Post result testing only the complete dataset transferred to the repository in the form of patch management or versioning. Finally, the appropriate/ supervised dataset is ready for synchronization to any existing VPN infrastructure along with the notification.

The few points processed to accomplish the Framework for Next Gen. VPN are as follows:

1. Overview of the Relevant Technologies.
2. Literature review and technical study of VPN, IPv6, Botnets, and other relevant andrequired technologies.
3. Collection of review and resources for finalization and implementation purposes.
4. Deployment of the basic framework and necessary infrastructure.
5. Implementation of the client Identity management and binding with MAC address andembed in IPv6.
6. Implementation of DHCP server with IPv6 and other encryption services.
7. Identify Challenges and limitations in the specific domains.
8. Customization of the network packets to place the client's identity in the packets
9. Identify and mitigate the Vulnerabilities in VPN connections using IP spoofing andother attacks.
10. Identify the challenges and Opportunities of using IPv6 for VPN connections

11. Behavior analysis of Botnets.
12. Implementation of End-to-End encryption and (2FA) Two Factor Authentication
13. Implementation of VPN-IPSec with HTTPS protocols for security reasons.
14. Evaluation of results and identify the benefits with comparison to the existingtechnologies.
15. Implemented centralized log server for the collection of all types of activities likesystem, users, for behavior analysis.
16. Applied Machine learning and customize algorithms for the use of testing methodologyon the framework based on critical parameters.
17. Updated information stored in database and repository to train the system for furtheractions to be taken by the Firewall, IDPs, etc.
18. Overall identify the findings and concluded thesis.
19. Identification of future scope to produce new standards in terms of secure networkcommunications.
20. Generate logs, Claims, and Disclaimers.

5.16 SCOPE FOR FUTURE STUDY

Because of the constantly changing IT security and related threats often based on computer and system security shield defects, further research is needed on this topic.

It is not an unknown or undocumented topic for botnets and its general infrastructure, although most malware contains the same basic code as many known botnets, minor modifications of infillsand spreading make it harder to secure them against. A more advanced tool is needed to combat the use of botnets and malware. Anti-malware tools are frequently based on code signatures and need a signature to identify the code as a threat.

VPN and Packet Filtration are already used separately to protect IPv6 personal data and computernetworks from the above attack techniques MITM and Smurf. Theoretically, it is possible to implement VPN and packet filtering jointly to ensure enhanced security, data protection, and costand improve network performance.

Major log reports and database servers must be obtained. The information could easily be obtainedfrom the specified servers, depending on the requirements. As it is continuous process management, it requires specific timestamps.

REFERENCES

1. Alhassoun, Manal M. ,(2016). “A Survey of IPv6 Deployment”, “(IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 7, No. 9, 2016, Page 42 - 46”
2. Anwar, Shahid (2018). “Android Botnets: A Serious Threat to Android Devices”, “Pertanika J. Sci. & Technol. 26 (1): 37 - 70 (2018) SCIENCE & TECHNOLOGY Journal, ISSN: 0128-7680 © 2018 Universiti Putra Malaysia Press.”
3. BALU J., (2014). “SECURE DATA TRANSMISSION OVER WIMAX NETWORKS USING VPN TECHNOLOGY IN REALTIME ENVIRONMENTS”, “IJCSMC, Vol. 3, Issue. 2, February 2014, pg.202 – 211”
4. Cert-IN, (2017). “Mirai Botnet Affecting IoT Devices”, [online] Available at: <https://www.cyberswachhtakendra.gov.in/alerts/mirai.html> [Accessed 08th October 2020]
5. Cert-IN, (2018). “Andromeda Botnet”, [online] Available at: <https://www.cyberswachhtakendra.gov.in/alerts/andromeda.html> [Accessed 08th October 2020]
6. Cert-IN, (2019). “Necurs Botnet”, [online] Available at: <https://www.cyberswachhtakendra.gov.in/alerts/Necurs.html> [Accessed 08th October 2020]
7. CHEN Jianqun, ZHANG, Haiyan (2009). “METHOD, DEVICE AND SYSTEM FOR PROTECTING MULTI-PROTOCOL LABEL SWITCHING RING NETWORK”, [online] Available at: < https://patentscope.wipo.int/search/en/detail.jsf?docId=WO2011038627 >, [Accessed 04, Nov. 2020]

8. Chigozie-Okwum C .C, (2019). “Botnet Identification Using Machine Learning Techniques: A Survey”, “2 nd International Conference on Education and Development ITED 2019, Page 1 - 12”

9. Cisco, (2019). “Connecting Remote Offices by Setting Up VPN Tunnels”, [online] Available at: <https://www.cisco.com/c/dam/en/us/td/docs/routers/csbr/app_notes/rv0xx_g2gvpn_an_OL-26286.pdf> [Accessed 18, December 2019]

10. Cisco.com, (2008). “Frame Relay”, Cisco. [online] Available at: <https://www.cisco.com/c/en/us/support/docs/wan/frame-relay/14193-frbacktoback.html , [Accessed 04, Nov. 2020]

11. Cristofaro De, Emiliano & Du, Honglu & Freudiger, Julien & Norcie, Greg. (2013). “Two-Factor or not Two-Factor a Comparative Usability Study of Two-Factor Authentication. USEC.”, doi: 10.14722/usec.2014.23025. Available at: <https://www.researchgate.net/publication/256846548_Two-Factor_or_not_Two-Factor_A_Comparative_Usability_Study_of_Two-Factor_Authentication > [Accessed 16, Dec., 2020]

12. David Leon Clark, (1999). “IT manager's guide to virtual private networks”, New York: McGraw – Hill – 1999. ISBN 0-07-135202-3, Available at: https://www.philadelphia.edu.jo/newlibrary/pdf/file7762e3e71b9e472ca232f46aa4f51946.pdf [Accessed 04, Nov. 2020]

13. Deering S., Hinden R. (1998). “Internet Protocol, Version 6 (IPv6) Specification”, Available at: https://datatracker.ietf.org/doc/html/rfc2460> [Accessed 14, December 2020]

14. en.wikipedia.org, (2019). ”Authentication Server”, [online] Available at: https://en.wikipedia.org/wiki/Authentication_server>” [Accessed 22, October,2019]

15. en.wikipedia.org, (2020). “Botnet”, [online] Available at < https://en.wikipedia.org/wiki/Botnet> [Accessed 29, October, 2020]

16. En.wikipedia.org, (2020). “Bridging”, [online] Available at: https://en.wikipedia.org/wiki/Bridging_(networking)> [Accessed 14, December 2020]

17. en.wikipedia.org, (2020). “End to End Encryption”, [online] Available at: https://en.wikipedia.org/wiki/End-to-end_encryption> [Accessed 11, June 2020]

18. en.wikipedia.org, (2020). “Identity Management”, [online] Available at: https://en.wikipedia.org/wiki/Identity_management > [Accessed 11, June 2020]

19. En.wikipedia.org, (2020). “IKE - Internet Key Exchange”, [online] Available at: https://en.wikipedia.org/wiki/Internet_Key_Exchange> [Accessed 21, October 2020]

20. En.wikipedia.org, (2020). “IPsec - Internet Protocol Security”, [online] Available at: https://en.wikipedia.org/wiki/IPsec> [Accessed 21, October 2020]

21. en.wikipedia.org, (2020). “IPv4”. [online] Available at: < https://en.wikipedia.org/wiki/IPv4>, [Accessed 29, October, 2020]

22. en.wikipedia.org, (2020). “IPv6”. [online] Available at: <https://en.wikipedia.org/wiki/IPv6> [Accessed 29, October, 2020]

23. En.wikipedia.org, (2020). “MAC address”, [online] Available at: https://en.wikipedia.org/wiki/MAC_address> [Accessed 04, Nov. 2020]

24. en.wikipedia.org, (2020). “Machine Learning”, [online] Available at: https://en.wikipedia.org/wiki/Machine_learning> [Accessed 15, Nov. 2020]

25. en.wikipedia.org, (2020). “National Science Foundation”, [online] Available at: <https://en.wikipedia.org/wiki/National_Science_Foundation> [Accessed 04, Nov. 2020]

26. En.wikipedia.org, (2020). “Neighbor Discovery Protocol”, [online] Available at: https://en.wikipedia.org/wiki/Neighbor_Discovery_Protocol > [Accessed 04, Nov. 2020]

27. En.wikipedia.org, (2020). “Network Time Protocol”, [online] Available at: https://en.wikipedia.org/wiki/Network_Time_Protocol> [Accessed 14, December 2020]

28. En.wikipedia.org, (2020). “Qualitative research”, [online] Available at: https://en.wikipedia.org/wiki/Qualitative_research> [Accessed 21, October 2020]

29. En.wikipedia.org, (2020). “SNMP - Simple Network Management Protocol”, [online] Available at: < https://en.wikipedia.org/wiki/Simple_Network_Management_Protocol> [Accessed 14, December 2020]

30. En.wikipedia.org, (2020). “SSH - Secure Shell Protocol”, [online] Available at: https://en.wikipedia.org/wiki/Secure_Shell_Protocol> [Accessed 14, December 2020]

31. en.wikipedia.org, (2020). “Virtual Private Network”, [online] Available at https://en.wikipedia.org/wiki/Virtual_private_network> [Accessed 29, October, 2020]

32. en.wikipedia.org, (2020). “Wide Area Network”, [online] Available at https://en.wikipedia.org/wiki/Wide_area_network> [Accessed 29, October, 2020]

33. en.wikipedia.org, (2020). “Zombie Computing”, [online] Available at: https://en.wikipedia.org/wiki/Zombie_%28computing%29> [Accessed 22, October, 2019]

34. Geethamani G.S, (2018). “A Review on Secure VPN using Network IPV6 based Moving Target Defense”, “International Journal for Research in Applied Science & Engineering Technology

35. (IJRASET) ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887 Volume 6 Issue III, March 2018- Available at www.ijraset.com”

36. Girdhar Anup, (2010). “Media Access Control (MAC) SPOOFING, A Biggest Threat For Cyber Crime Investigation.”, Information Management in Knowledge

Economy, ISBN 10:0230-32936-5, ISBN 13: 978-0230-32936-2. , “Macmillan Publishers India Ltd.”, pp 227 – 237, (March , 2010)

37. Giri Kartik, Saxena Namit, Srivastava Yash, Saxena Pranshu, (2020). “End-to-End Encryption Techniques”, International Research Journal of Engineering and Technology (IRJET), Volume: 07 Issue: 06 (June 2020), e-ISSN: 2395-0056, p-ISSN: 2395-0072
38. Guard (2019). “Understanding and addressing Android VPN vulnerabilities”, [online] Available at: <https://www.insidesecure.com/Media/Files/Whitepapers/Understanding-and-addressing-Android-VPN-vulnerabilities> [Accessed 18, October 2019]
39. Gurtov A, (2008). “Host Identity Protocol (HIP)”, “Towards the Secure Mobile Internet”, Wiley Series on Communications Networking & Distributed Systems. “Wiley, (2008)”
40. Hadianto, Rahmadani, (2018). “A Survey Paper on Botnet Attacks and Defenses in Software Defined Networking”, “International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 1 (2018) pp. 483-489 © Research India Publications. http://www.ripublication.com”
41. Hanks Stanley, Farinacci Dino, Li Tony, Traina Paul. (2014). “Generic Routing Encapsulation (GRE)”, RFC 1701, [online] Available at: < https://datatracker.ietf.org/doc/rfc1701/> [Accessed 04, Nov. 2020]
42. Heydarian, Mohsen (2012). “A high performance optimal dynamic routing algorithm with unicast multichannel QoS guarantee in communication systems”, Springer (springer.com), [online] Available at: < https://link.springer.com/article/10.1007/s11227-011-0723-0 > [Accessed 04, Nov. 2020]
43. ICCSA 2006: Computational Science and Its Applications - ICCSA 2006 pp 486-496”
44. Ietf.org, (2007), “IPv6 Stateless Address Autoconfiguration”, [online] Available at: < https://datatracker.ietf.org/doc/html/rfc4862> [Accessed 14, December 2020]

45. Ikram, Muhammad (2016). “An Analysis of the Privacy and Security Risks of Android VPN Permission-enabled Apps”, “ACM. ISBN 978-1-4503-4526-2/16/11”.

46. JAYANTHI GOKULAKRISHNAN, (2014). “A SURVEY REPORT ON VPN SECURITY & ITS TECHNOLOGIES”, “Jayanthi Gokulakrishnan et.al / Indian Journal of Computer Science and Engineering (IJCSE), ISSN : 0976-5166, Vol. 5 No.4 Aug-Sep 2014, Page 135 - 139”

47. Jyothi, K. Karuna, (2018). “Study on Virtual Private Network (VPN), VPN’s Protocols And Security”, “International Journal of Scientific Research in Computer Science, Engineering and Information Technology © 2018 IJSRCSEIT | Volume 3 | Issue 5 | ISSN : 2456-3307”

48. Kang Byeong-Ho, (2009). “Vulnerabilities of VPN using IPSec and Defensive Measures”, “International Journal of Advanced Science and Technology Volume 8, July, 2009”, “Page 9 – 18”.

49. Khan Mohammad Taha, (2018), “An Empirical Analysis of the Commercial VPN Ecosystem”, “ACM ISBN 978-1-4503-5619-0/18/10”

50. Lim, Hyung-Jin, (2006). “Comparative Analysis of IPv6 VPN Transition in NEMO Environments”, ” Springer, International Conference on Computational Science and Its Applications

51. Linda Rosencrance, Peter Loshin, Michael Cobb, (2020). “two-factor-authentication”, [online] Available at: <https://searchsecurity.techtarget.com/definition/two-factor-authentication> [Accessed 04, Nov. 2020]

52. Linda Rosencrance, Peter Loshin, Michael Cobb, (2020). “two-factor-authentication”, [online] Available at: <https://searchsecurity.techtarget.com/definition/two-factor-authentication> [Accessed 04, Nov. 2020]

53. Meta Networks, (2018). “MetaNetworks-Whitepaper-SDP-vs-VPN”, [online] Available at: <https://www.metanetworks.com/wp-

content/uploads/2018/12/MetaNetworks-Whitepaper-SDP-vs-VPN.pdf> [Accessed 18, October 2020]

54. Musthaler Linda, (2019). “Tempered networks simplifies secure network connectivity and microsegmentation” [online] Available at: < https://www.networkworld.com/article/3405853/tempered-networks-simplifies-secure-network-connectivity-and-microsegmentation.html> [Accessed 14, December 2020]
55. NCSC, (2019). “Vulnerabilities exploited in VPN products used worldwide”, [online] Available at <https://www.ncsc.gov.uk/news/alert-vpn-vulnerabilities> [Accessed 11th Nov. 2020]
56. networkencyclopedia.com, (2020). “Frame Relay” Available at: < https://networkencyclopedia.com/frame-relay/>, Available at: [Accessed 04, Nov. 2020]
57. NIC, (2019). “NIC VPN Policy”, [online] Available at: <https://vpn.nic.in/resources/application_forms/NIC-VPN-policy-V3-draft.pdf> [Accessed 18, December 2020]
58. Openvpn.net, (2020). “OpenVPN – Secure Networking”, [online] Available at: < https://openvpn.net/> [Accessed 21, October 2020]
59. Perta, Vasile C., (2015). “A Glance through the VPN Looking Glass: IPv6 Leakage and DNS Hijacking in Commercial VPN clients”, “De Gruyer open, Proceedings on Privacy Enhancing Technologies 2015; 2015 (1):77–91”
60. Powell, J. Myles, (2010). “The Impact of Virtual Private Network (VPN) on a Company 's Network”, “”Utah State University DigitalCommons@USU, Spring 2010”
61. Questionpro.com, (2020). “Research - What is Research: Definition, Methods, Types & Examples”, [online] Available at: < https://www.questionpro.com/blog/what-is-research/ > [Accessed 16, Dec., 2020]

62. Ramadhani E. (2018). “Anonymity communication VPN and Tor: a comparative study”, “IOP Conf. Series: Journal of Physics: Conf. Series 983 (2018) 012060 doi :10.1088/1742-6596/983/1/012060”

63. Sanaz Rahimi, (2017). “Security Analysis of VPN Configurations in Industrial Control Environments”, “Hal archives-ouvertes.fr, 5th International Conference Critical Infrastructure Protection (ICCIP), pp.73-88, ff10.1007/978-3-642-24864-1_6ff. ffhal-01571782f”.

64. Sekar, N., (2017). “AN EMPIRICAL STUDY ON INTERNET PROTOCOL IPV6 IN NETWORKING”, Thomsan Reuters, INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY, ISSN: 2277-9655, Page 135 – 138”.

65. Sharma Tina, (2013). “Statistical Results of IPSec in IPv6 Networks”, “International Journal of Computer Applications (0975 – 8887) Volume 79 – No.2, October 2013, Pages 14 - 19”

66. Sharma Viney, (2010). “IPv6 and IPv4 Security challenge Analysis and Best-Practice Scenario”. Int. J. of Advanced of Networking and Applications, Volume: 01, Issue: 04, Pages: 258-269 (2010). Available at: < http://www.ijana.in/papers/4.9.pdf > [Accessed 04, Nov. 2020]

67. Sosinsky, Barrie, (2009). “Network Bible”, John Wiley & Sons, 2009, ISBN 0470543426, 9780470543429. PP – 341.

68. Srinidhi K S, (2014). “Tunnel-based ipv6 transition with automatic bandwidth management”, “International Journal of Computer Science and Mobile Computing, ISSN 2320–088X, IJCSMC, Vol. 3, Issue. 6, June 2014, pg.360 – 366”

69. Stokes, Prof. Andrew (2017). “Sampling, Sample Size and Planning for Data Collection”, [online] Available at: <http://sites.bu.edu/gh811/files/2017/10/Wk_4.pdf> [Accessed 16, Dec., 2020]

70. Tambe Amit, (2019). “Detection of Threats to IoT Devices using Scalable VPN-forwarded Honeypots”, “cyxtera.com”

71. Tayal, Dr. Sandeep, (2017). “A Review paper on Implementation Issues in IPv6 Network Technology”, “International Journal of Electronics Engineering Research. ISSN 0975-6450 Volume 9, Number 4 (2017) pp. 491-498, Page 491 - 498”

72. Techtarget.com, (2020). “ARPANET” [online] < https://searchnetworking.techtarget.com/definition/ARPANET> [Accessed 04, Nov. 2020]

73. Thanh Bui, (2019). “Client-side Vulnerabilities in Commercial VPNs”, [online] Available at: <https://arxiv.org/pdf/1912.04669.pdf> [Accessed 16, Dec., 2019]

74. Thiruvasagam, Prabhu (2019). “IPSec: Performance Analysis in IPv4 and IPv6”, “Journal of ICT, Vol. 7 1, 59–76. River Publishers doi: 10.13052/jicts2245-800X.714”

75. Thompson, Andrew (2017). “A Critical Evaluation of Current Research into Improving Botnet Detection Rates”, ” Kendal, Simon (2017) Selected Computing Research Papers Volume 6 June 2017. Selected Computing Research Papers . University of Sunderland, Sunderland., Page 55 - 60”

76. Worster T., Rekhter Y., Rosen, E. Ed., (2005). “Encapsulating MPLS in IP or Generic Routing Encapsulation (GRE)”, RFC 4023, [online] Available at: < https://datatracker.ietf.org/doc/html/rfc4023> [Accessed 04, Nov. 2020]

77. Yadigiri, Vorsu (2017). “A Survey on Botnet Detection Based On Anomaly and Community Detection”,” International Journal of Innovative Research in Science, Engineering and Technology, Vol. 6, Issue 3, March 2017, ISSN(Online) : 2319-8753 ISSN (Print) : 2347-6710”

78. ZAHARIA, ANDRA. (December 2020). “Cybercrime and Cybersecurity Statistics & Trends”, [online] Available at: <https://www.comparitech.com/vpn/cybersecurity-cyber-crime-statistics-facts-trends/> [Accessed 18, December 2020]

79. Zhang Ting Ting, (2012), “A Research on IPv6/IPv4-Based Network Performance Test”, “ Science Direct, Procedia Engineering, Volume 29, 2012, Pages 1573 - 1577”

80. Zheng, L. Ed., Penno R., Wang Z. (2016). “Yang Data Model for Generic Routing Encapsulation (GRE)”, [online] Available at: < https://datatracker.ietf.org/doc/html/draft-zheng-intarea-gre-yang-01 > [Accessed 04, Nov. 2020]

WEB SITES

1. https://restoreprivacy.com/vpn/best/ipv6/
2. https://www.itopvpn.com/blog/ipv6-vpn-1430
3. https://vpnalert.com/best-virtual-private-network/ipv6-support/
4. https://docs.aws.amazon.com/vpn/latest/s2svpn/ipv4-ipv6.html

REFRENCE BOOKS

- CCNA 200-301 Portable Command Guide
- Network Analysis Using Wireshark 2 Cookbook
- Understanding Cisco Networking Technologies, Volume 1
- CCNP Enterprise Advanced Routing ENARSI 300-410 Official Cert Guide
- IP Subnetting

SURVEY :

1. Name

……..……..…..

2. Email ID

…….……

3. Phone Number

……..……

4. Designation

…….……..….

5. Organization Name

……..….

6. Location
(mark only one box)

- ☐ New Delhi
- ☐ Pune
- ☐ Bengaluru
- ☐ Gurugram
- ☐ Mumbai
- ☐ Noida
- ☐ Hyderabad
- ☐ Kerala
- ☐ Other: …….....….

7. Type of Organization

- ◯ Telecommunication
- ◯ Information Technology
- ◯ Education
- ◯ Financial/Banking
- ◯ Healthcare/Medical
- ◯ Government
- ◯ Retail
- ◯ Transportation
- ◯ Other: ……..

8. Degree of Responsibility towards IT Infrastructure/ Data Security in your Organization?

- ☐ Primary Responsibility
- ☐ Secondary Responsibility
- ☐ Responsible when required
- ☐ No Responsibility

9. Is Remote User Authentication allowed in your Intranet based Application Servers?

- ☐ Yes
- ☐ No
- ☐ On Demand

10. Is IPv6 Implemented in your organization?

- ☐ Yes
- ☐ No
- ☐ On Demand

11. Is Official VPN configured in your organization?

- [ ] Yes
- [ ] No
- [ ] On Demand

12. Do you maintain End User Trusted Devices in your Organization?

- [ ] Yes
- [ ] No
- [ ] On Demand

13. Do you have Hardware Level Firewall in your organization?

- [ ] Yes
- [ ] No
- [ ] On Demand

14. What percent of overall budget is allocated in your company to Cyber Security?

- [ ] <10%
- [ ] 10%-20%
- [ ] 20%-30%
- [ ] 30%-40%
- [ ] 40%-50%
- [ ] >50%

15. What type of Devices/ Platform is normally connected to your organizational Network?

- [ ] Windows
- [ ] Linux
- [ ] Apple OS
- [ ] Android
- [ ] All of the above

16. What is the level of Cyber Security implemented in your Organization?

- [ ] High
- [ ] Moderate
- [ ] Low
- [ ] No Concern

17. What is the approach to identify the legitimate devices connected to the Network?

- [ ] Through IP Address
- [ ] Through Log Server
- [ ] Through Firewall
- [ ] Through Mac Address
- [ ] Through User Credentials

18. What is the average response time for Incident Handling?

- [ ] <1 hour
- [ ] 1 day
- [ ] >1 day
- [ ] 1 week
- [ ] >1 week

19. What is the important factor of Cyber Security Challenges in your organization?

- [ ] User Authentication and Granting Privileges
- [ ] Malware Attacks
- [ ] End Point Security
- [ ] Intranet Security
- [ ] Miscellaneous Threats

20. Level of Confidence in terms of Cyber or Network Security In your organization?

- [ ] <10%
- [ ] 10%-20%
- [ ] 20%-30%
- [ ] 30%-40%
- [ ] 40%-50%
- [ ] >50%