

Penetration Testing IPsec VPNs

Rohyt Belani, K. K. Mookhey 2005-02-09

1. Introduction

As companies expand their presence globally, there arises a need for secure electronic communications between geographically dispersed locations. Virtual private networks (VPNs) provide an economically viable option to address this need. A VPN is a private network that uses the public Internet to either connect remote users to the company's internal network or establish a seamless connection between the company's physically isolated sites. Since a VPN uses the Internet it must provide security features like encryption and strong authentication to protect the confidentiality of internal company data. Thus there are inherent security concerns when implementing VPNs.

This article discusses a methodology to assess the security posture of an organization's Ipsec based VPN architecture. The first part of the article looks at the components of IPsec based VPNs, which use client software to connect to the VPN server as opposed to SSL based VPNs, which only use a browser. The second step describes a penetration test of the VPN setup, and then finally a review of the architecture and system configuration is suggested. A comprehensive VPN assessment must account for all possible attack vectors for it to be a useful gauge of security posture.

2. Components of IPsec based VPNs

VPNs can be classified into two primary types. Site-to-Site VPNs virtually extend the corporate LAN to a company's satellite offices, and this connectivity is established between VPN gateways at each participating location. Remote Access VPNs, on the other hand, are used to provide a remote user, such as a field sales person, access to the internal corporate network from various remote locations. The two primary components that comprise this user-to-LAN connection are a VPN client software that operates on the remote user's machine and a VPN gateway that is the entry point into the company's internal network from the Internet. We will look at layer-3 IPsec VPNs that require a thick VPN client, as opposed to SSL-based VPNs that require only a browser on the client machine to establish connectivity to internal resources like file and mail servers.

IPsec is based on symmetric-key encryption and consists of the following primary security components:

1. **Authentication Header (AH):** This is essentially a message authenticity checksum that is appended to every packet to ensure its authenticity and protect its integrity as it traverses the Internet.
2. **Encapsulating Security Payload (ESP):** This is the encryption mechanism used to protect the confidentiality of communication between the subjects.
3. **Internet Key Exchange (IKE):** This protocol provides a means to securely exchange the secret key, which is essential for the effective operation of the AH and ESP between the communicating

subjects. While the secret keys can be manually exchanged, such a solution is not scalable and the keys should be changed periodically to minimize the probability of their compromise. IKE has two main modes - IKE Main Mode and IKE Aggressive Mode. Main Mode key-exchange uses the Diffie-Helman exchange to generate a mutual shared key between the client and the server. On the other hand, Aggressive Mode does not use a Diffie-Helman exchange to protect the authentication data. Therefore, it is possible to capture this authentication data using a sniffer, and crack it offline.

3. Penetration test

The main objective of this phase is to discover any vulnerabilities in the VPN implementation that an attacker may be able to exploit. This is usually considered a zero-knowledge test where only the IP address of the VPN server is known. This phase will be shown using three steps:

1. Reconnaissance: determining open ports and doing VPN fingerprinting
2. Assessment of PSK protocol mode
3. Exploitation of any default user accounts

3.1 Reconnaissance

The purpose of this exercise is to determine the type of VPN implementation (IPsec, PPTP, or SSL), the VPN vendor information and corresponding version numbers. This is necessary to execute a focused attack against the target VPN environment.

The first step in thereconnaissance process entails port scanning the VPN server to make an educated guess on the type of VPN implementation. The following table provides a mapping of open ports to VPN type, using default ports:

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

Table 1. Default ports and VPN type.

It is possible that the port scan may yield false positives. This is most likely to happen if the subject of the scan is a firewall-VPN combination device. In such cases, the firewall is likely to drop packets targeted to it, resulting in false positives.

The next step is to determine what we are up against by finding out the vendor and version of the VPN server. Nmap's operating system identification functionality provides a fair idea of the above. In the case of IPsec VPNs, [ike-scan](#) can also be used to provide a reasonably accurate fingerprint of the VPN server vendor

and the version number. This tool performs the fingerprinting by checking the values of specific variables in the IPsec packets being exchanged, and compares these against its signature database.

The following is a snapshot of the execution of IKE-Scan:

```
G:\ike-VPN-test>ike-scan 10.0.0.1
Starting ike-scan 1.6 with 1 hosts (http://www.nta-monitor.com/ike-scan/)
10.0.0.1 Main Mode Handshake returned SA=(Enc=3DES Hash=SHA1 Auth=PSK Group=1:
modp768 LifeT
ype=Seconds LifeDuration(4)=0x00007080)

Ending ike-scan 1.6: 1 hosts scanned in 0.979 seconds (1.02 hosts/sec).
1 returned handshake; 0 returned notify
```

If the packet exchange patterns observed by ike-scan match any of those within its signature database, it provides a best guess of the VPN server platform and outputs the data as follows:

```
Implementation guess: Netscreen
Or
Implementation guess: Cisco IOS/PIX
```

The fingerprint information allows a potential attacker the ability to execute a more focused attack against the VPN. With this knowledge, the attacker may not only attempt platform specific exploits against the system, but may also attempt a brute-force attack using the appropriate client software.

3.2 Assessment of PSK protocol mode

Another vital attack vector is the exploitation of inherent vulnerabilities in the protocols used to establish the VPN connection. For instance, the output from ike-scan above shows us that the client and server use a pre-shared key (PSK) for authentication. Thus, if the server can be forced to use Aggressive Mode, instead of Main Mode authentication, then the authentication hash based on the pre-shared key (PSK) would be sent in clear-text. From there it is then possible to use a sniffer like tcpdump to capture the hash and attempt a dictionary or brute-force attack against it to recover the PSK.

[IKEProbe](#) can be used to determine vulnerabilities in the PSK implementation of the VPN server. It tries out various combinations of ciphers, hashes and Diffie-Helman groups and attempts to force the remote server into aggressive mode.

The following is a snapshot of IKEProbe output:

```
IKEProbe 0.1beta (c) 2003 Michael Thumann (www.ernw.de)
Portions Copyright (c) 2003 Cipherica Labs (www.cipherica.com)
Read license-cipherica.txt for LibIKE License Information
IKE Aggressive Mode PSK Vulnerability Scanner (Bugtraq ID 7423)
```

Supported Attributes

```
Ciphers          : DES, 3DES, AES-128, CAST
Hashes           : MD5, SHA1
Diffie Hellman Groups: DH Groups 1,2 and 5
```

```
IKE Proposal for Peer: 10.0.0.2
Aggressive Mode activated ...
```

[Output truncated for brevity]

Cipher AES

Hash MD5

Diffie Hellman Group 2

```
841.890 3: ph1_initiated(00443ee0, 007d23c8)
841.950 3: << ph1 (00443ee0, 276)
843.963 3: << ph1 (00443ee0, 276)
846.967 3: << ph1 (00443ee0, 276)
849.961 3: ph1_disposed(00443ee0)
```

Attribute Settings:

Cipher AES

Hash MD5

Diffie Hellman Group 5

```
849.961 3: ph1_initiated(00443ee0, 007d5010)
849.141 3: << ph1 (00443ee0, 340)
851.644 3: << ph1 (00443ee0, 340)
854.648 3: << ph1 (00443ee0, 340)
857.652 3: ph1_disposed(00443ee0)
```

[Output has been truncated]

The sniffed PSK can be cracked using [Cain & Abel](#) or [IKECrack](#). Once the PSK has been cracked, software such as PGPNet can be used to connect to the vulnerable VPN server. This is explained by Micheal Thumann (author of ike-probe) in his [PSK cracking paper](#) (PDF).

An attacker may also attempt to exploit vulnerabilities in the vendor's implementation of the specific protocols. As Bruce Schneier and N. Ferguson wrote in their paper, "[A Cryptographic Evaluation of IPsec](#)," the main problem with the IPsec protocol is its inherent complexity. Therefore, there is a strong likelihood of vendors developing buggy implementations of the protocol which are vulnerable to attack.

The following is a list of well-known VPN/IKE vulnerabilities from the SecurityFocus Vulnerabilities database. There are likely others as well that have not been reported yet.

- [Nortel Contivity VPN Client Username Enumeration Vulnerability](#)
- [Nortel Contivity VPN Client Gateway Certificate Check Failure Vulnerability](#)
- [Cisco IOS Malformed IKE Packet Remote Denial Of Service Vulnerability](#)
- [Multiple Vendor IKE Implementation Certificate Authenticity Verification Vulnerability](#)
- [Multiple Vendor IKE Insecure XAUTH Implementation Vulnerabilities](#)
- [Check Point VPN-1 IKE Aggressive Mode Forcing Vulnerability](#)
- [Netscreen-Remote VPN Client IKE Packet Excessive Payloads Vulnerability](#)
- [PGPfreeware Malformed IKE Response Packet Buffer Overflow Vulnerability](#)
- [Cisco VPN Client Zero Length IKE Packet Denial Of Service Vulnerability](#)
- [Cisco VPN Client IKE Security Parameter Index Payload Buffer Overflow Vulnerability](#)
- [Cisco VPN Client IKE Packet Excessive Payloads Vulnerability](#)
- [OpenBSD isakmpd IKE Payloads Denial Of Service Vulnerability](#)

The first of these, for instance, was discovered by the author during a routine penetration test of a client's VPN server. As shown below, when connecting with the VPN client of a Nortel Contivity Server, the responses displayed for a valid username (but invalid password) is different than that for an invalid username.

The first screenshot, Figure 1, uses the valid username *test* and shows the following error:



Figure 1. Nortel VPN client with valid username.

However, if we use an account such as *test123*, which does not exist, the error message shown is:

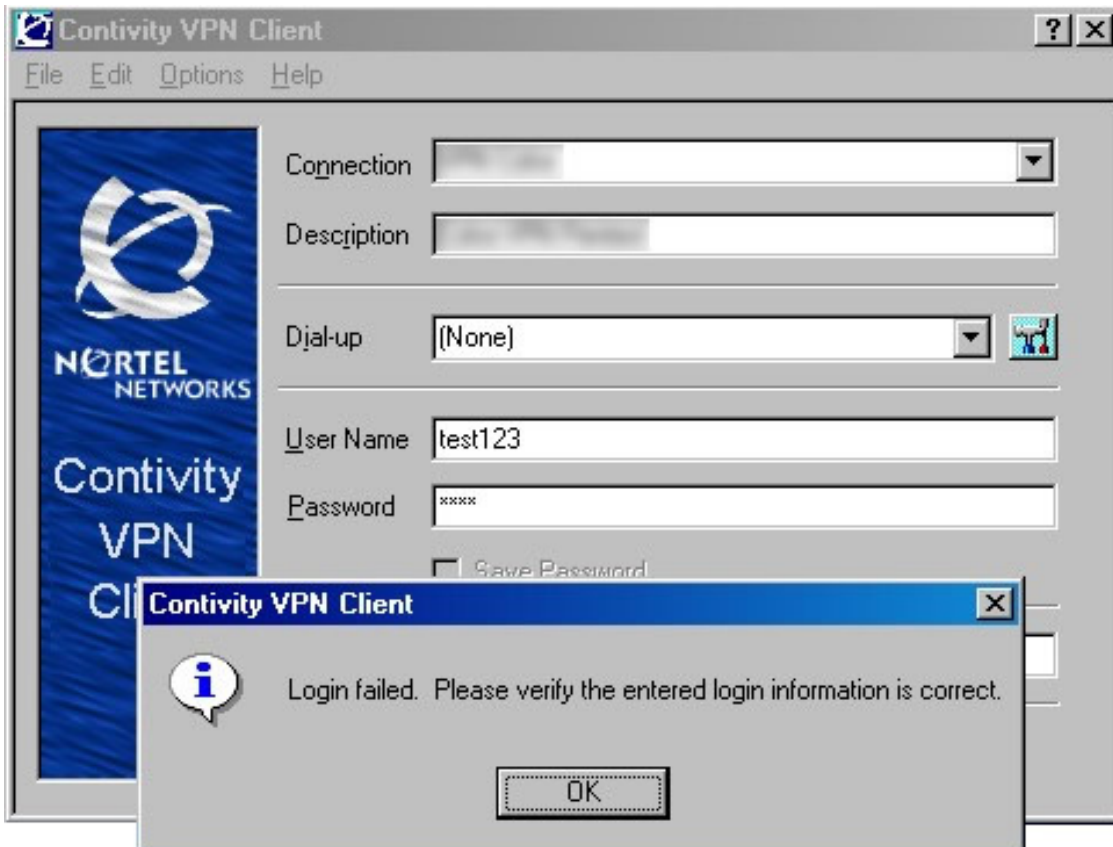


Figure 2. Client shows username does not exist.

This vulnerability may be used by an attacker to enumerate usernames of valid VPN accounts.

3.3 Exploitation of any default user accounts

One of the common vulnerabilities in the implementation of any system is the presence of default system accounts with default passwords. VPN systems are no exception. A good source of default account names and passwords can be [found](#) through various means. Besides the usual suspects such as *vendor-name*, *setup*, *vpn*, *client*, *user*, *contivity*, *fw1*, *netscreen*, and *admin*, the assessor should also try the names of the cities/towns where the remote offices are located. It is not uncommon for remote office users from the London office, for example, to have the username *london*!

In addition to blind penetration testing (without a valid user account), assessing the VPN using a valid user account ID provides added value. This normally yields a larger number of critical vulnerabilities than the blind penetration testing phase. This can be attributed to the added VPN functionality and attack surface exposed to an authenticated user as compared to a zero-knowledge attacker. The primary focus of this phase is to ensure that the extent of access granted to VPN users is limited as per stipulated corporate policies.

4. Configuration and architecture review

To perform a thorough VPN assessment it is critical that one review the network architecture and configuration of the VPN. Some of the issues that should be evaluated are:

1. The kind of access has been given to authenticated VPN users, and whether this has been restricted to specific servers and ports within the internal network.
2. Whether two-factor authentication, like RSA SecureID, is being used. If not, there should be a business case to justifying the lack of use of such an important mechanism.
3. Whether the VPN server has been securely configured to disallow aggressive mode authentication combined with pre-shared keys (PSK). If aggressive mode is required, it must be used with digital certificates or some other form of two-factor authentication to strengthen the authentication.
4. Whether only required accounts have been created on the system, and whether each user has the correct authorization levels. Generic accounts must not be used, and each user has a single, distinct account.
5. Whether split tunneling is disabled on the VPN clients. Split tunneling allows one to configure specific network routes on client to go through the tunnel, while any other traffic goes to the local PC interface. Disabling this capability prevents a random Internet-based attacker from compromising the VPN client machine when it is connected to corporate network over the Internet.
6. And finally, ensure that all necessary patches have been applied.

5. Conclusion

VPNs are commonly ignored during a vulnerability assessment, due to the myth that they are inherently secure. While VPNs do provide a means for secure communication, if they are incorrectly configured they are still vulnerable, just as any other Internet-facing system. The compromise of a VPN server may have an extremely negative impact on the organization's business as it may provide unauthorized access to internal company resources. Thus, organizations should pay special attention to the design, implementation, configuration and assessment of VPN systems, and ensure a proper penetration test has been completed.

About the authors

[Rohyt Belani](#) is a Director at Red Cliff Consulting, an information security company offering professional services targeting proactive and reactive security initiatives. [K. K. Mookhey](#) is the CTO and Founder of Network Intelligence India.

View more articles by [Rohyt Belani](#) or [K.K. Mookhey](#) on SecurityFocus.

[Privacy Statement](#)

Copyright 2006, SecurityFocus