GIAC
CERTIFICATIONS

Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permited without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"Security Essentials: Network, Endpoint, and Cloud (Security 401)"
at http://www.giac.org/registration/gsec

Securing Wireless Clients using IPsec via Linux Gateway
GSEC Practical Assignment v.1.4b (Option 1)

Robert B. King
June 27, 2003

Abstract:
This document describes how to configure Linux clients for communication with a Linux server via IPsec. We describe the use of wireless clients connecting to an internal private LAN (which is, in turn, connected to the Internet via a firewall). Background information on the software programs as well as on IPsec is discussed.

The latter part of the document illustrates the configuration information for FreeS/WAN. Detailed information on the installation of FreeS/WAN, its configuration, and the creation of X.509 certificates is also discussed.

1. Introduction

The use of wireless networking in residential and small office/home office (SOHO) is growing. Given that Wi-Fi is becoming more prevalent worldwide, with a growth rate projected to be fifty-seven percent annually over the next five years[1], usage of wireless networks will grow elsewhere as more and more computers are equipped with wireless network adapters. A distinct advantage of wireless networking is the ability to deploy networks without running cabling to connect the various computers and other devices (e.g., network printers, firewalls, routers). This reduces the cost for deploying these devices, particularly in the SOHO and small business environments where cost is important.

Existing 802.11b wireless networks rely on the Wired Equivalent Privacy (WEP) algorithm to protect wireless communications from eavesdropping and to prevent unauthorized users from utilizing the network. Borisov *et al.* presented a paper at the *7th Annual International Conference on Mobil Computing and Networking* where they described the flaws in WEP and techniques for attacking the algorithm.[2] They suggest that one should not rely on WEP to provide secure transmissions between wireless devices.[3]

Even with these flaws, some wireless network vendors do not enable WEP as the factory default. There are sites on the Web which list the default

---

[1] Pruitt, Scarlet. "Wi-Fi Faces Growing Pains." PCWorld.COM. URL: http://www.pcworld.com/news/article/0,aid,111362,00.asp (June 27, 2003).
[2] Borisov, Nikita, Ian Goldberg, and David Wagner. "Intercepting Mobile Communications: The Insecurity of 802.11". *7th Annual International Conference on Mobile Computing and Networking,* July 16-21, 2001. URL: http://www.isaac.cs.berkeley.edu/isaac/mobicom.pdf.
[3] "Security of the WEP algorithm." URL: http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

configurations of wireless products from various vendors[4]. From practical experience, I have found that all of the wireless networks that I detected recently when configuring a new network where unprotected, did not have WEP enabled, nor did they filter MAC addresses to prevent others from gaining access to their network.

Wireless networks which conform to the new Wi-Fi Protected Access (WPA) are starting to appear. WPA constructs its encryption keys differently from WEP – by using a Temporal Key Integrity Protocol (TKIP).[5] In essence, this utilizes dynamic encryption keys to mitigate the ease of generating the key from monitoring existing traffic.

This paper presents one technique for improving the security of wireless networks by utilizing IPsec to encrypt and authenticate traffic between the wireless clients, e.g., laptops, and a gateway. Other techniques are available, including OpenSSH, which also supports secure tunneling capabilities.[6] These methods provide techniques for encrypting and authenticating traffic over untrustworthy networks. We are aware of the new WPA standard which should allievate some need for such additional security. However, the focus of this paper is on SOHO networking configurations which continue to utilize their existing technologies for as long as possible. For that reason, utilizing Linux is particularly cost effective.

1.1 Network Overview

A SOHO location can have a network consisting of systems connected by both wireless and wired networks. From a security standpoint, most systems use firewalls to prevent unauthorized access to the SOHO network from the Internet. However, if a wireless network is also deployed at the site, the network administrator should consider deploying some sort of security mechanism to prevent unauthorized wireless network users from entering the network.

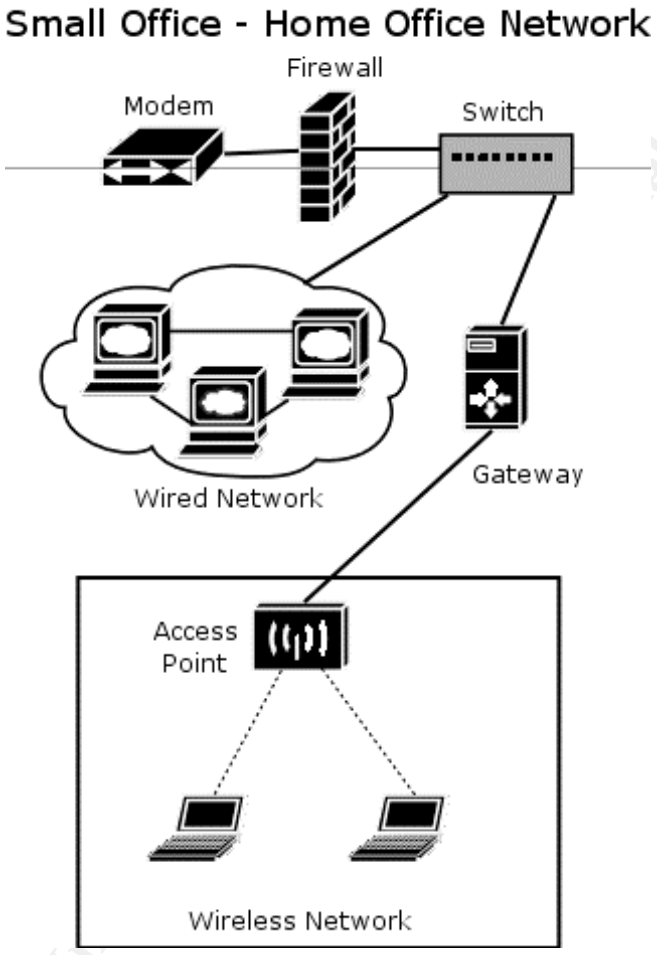
The basic network configuration is depicted in Figure 1. Connectivity to the Internet is obtained via a DSL or cable modem which is, in turn, connected to a firewall device. The firewall device is attached to a network switch (or hub) to which computers connected via Ethernet are attached. A wireless gateway routes traffic between the wired subnet and the wireless subnet. Attached to the wireless gateway is a wireless access pointer. Mobile devices, such as computer laptops, use wireless adapters to attach to this network.

---

[4] "Default Wireless Configurations." URL: http://www.cirt.net/cgi-bin/ssids.pl.
[5] Arar, Yardena. "Wi-Fi Networks Get a Security Upgrade." PCWorld.COM. April 30, 2003. URL: http://www.pcworld.com/news/article/0,aid,110520,00.asp.
[6] OpenSSH. URL: http://www.openssh.org.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Logically, this could represent a wired residence with desktops in various users bedrooms and offices. The wireless network corresponds to laptops which can be used from anywhere else in the house. In an office scenario, this could represent office workers using desktops, but then allow them to work from a lunchroom or elsewhere in the building.



In both scenarios, office and home, the range of wireless networks have been shown to exceed reported maximum distances. Given this and the relative lack of security of wireless networking, this paper will look at the wireless portion of the network further and a technique to provide additional security using FreeS/WAN on Linux to provide IPsec for traffic between mobile laptops running Linux and the Linux gateway. The corresponding configuration parameters for using Windows 2000/XP clients are outside the scope of this paper. The Linux gateway is only functioning as a router, it is not providing Network Address Translation (NAT) services. This is no a limiting factor because both the wireless subnet and wired subnet are Class C private subnetworks.

1.2 IPsec

Internet Protocol Security (IPsec) uses strong cryptography to provide both authentication and encryption services. By existing at the IP level of the network

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

stack, it is able "protect *any protocol* running above IP and *any medium* which IP runs over."[7] IPsec functions on a system level, providing many of its services in the background, with little or no impact to its users. Likewise, it authenticates the systems, not the users. Thus, one use of IPsec is to create secure tunnels through untrusted networks (such as wireless networks).[8]

IPsec utilizes three protocols:

- Encapsulating Security Payload (ESP) – encrypts and/or authenticates data using a block cipher. It utilizes IP protocol 50.[9]
- Authentication Header (AH) – provides a packet authentication service, but no encryption. It utilizes IP protocol 51.[10]
- Internet Key Exchange (IKE) – negotiates connection parameters (such as algorithms, keys, connection lifetime) and subsequently sets up either ESP or AH connections. It is also responsible for rekeying, repairing, and tearing down IPsec connections and utilizes UDP port 500.[11]

2. Software

Various software packages are utilized to implement the network structure. We discuss them briefly in this section.

2.1 Gentoo Linux

The wireless gateway system runs Gentoo Linux1.4 because it is easier to customize and optimize than more traditional distributions, such as Red Hat, SuSE, and Mandrake. . In server environments, it is extremely advantageous to only install the required software packages with the required options. This improves the security of the system[12]. Gentoo utilizes Portage[13] to provide to build applications from sources. By altering the USE parameter in the /etc/make.conf file, one can strictly control the interdependencies between the various applications and therefore reduce the number of services needed. With a distribution such as Red Hat, you can control what RPMs are loaded; however, if there are times that RPMs have requirements which cannot be met without installing RPMs that you do not need. With Gentoo, you can alter the USE parameter to minimize the number of times that this occurs. For instance, we can build a gateway server that has no X11, no KDE, and no GNOME support. All the applications included are command line-based.

---

[7] "The IPsec protocols." URL: http://www.freeswan.org/freeswan_trees/2.0/ipsec.html.
[8] Ibid.
[9] Kent, S. and R. Atkinson. "IP Encapsulating Security Payload (ESP)."The Internet Engineering Task Force. Novermber 1998. URL: http://www.ietf.org/rfc/rfc2406.txt?number=2406.
[10] Kent, S. and R. Atkinson. "IP Authentication Header." The Internet Engineering Task Force. November 1998. URL: http://www.ietf.org/rfc/rfc2402.txt?number=2402.
[11] Kent, S. and R. Atkinson. "The Internet IP Security Domain of Interpretation for ISAKMP." The Internet Engineering Task Force. November 1998. URL: http://www.ietf.org/rfc/rfc2407.txt?number=2407.
[12] Hughes, Woody. "Iptables, Linux and You!" GSEC Practical Assignment v.1.4b (Option 1). March 3, 2003. URL: http://www.giac.org/practical/GSEC/Woody_Hughes_GSEC.pdf
[13] Portage. URL: http://www.gentoo.org/doc/en/faq.xml

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

.

2.2 FreeS/WAN 2.0

In April 2003, the team which developed FreeS/WAN released version 2.0 which added built-in support for "opportunistic encryption", policy groups, packetdefault connection, disable reverse packet filtering, and updated ipsec.conf configuration file format. Opportunistic encryption, in conjunction with a secure name server to provide the appropriate keys, provides encrypted traffic, but not authenticated traffic. Note that FreeS/WAN configurations found on the Internet for versions prior to 2.0 may not work without some modifications to support the changes in 2.0.[14]

A patch to support X.509 or OpenPGP certificates can be obtained from strongSec and applied to FreeS/WAN. The X.509 patch allows RSA-based authentications using these certificates between a Linux FreeS/WAN gateway and IPsec clients. The certificates can be retrieved using X.509 services via the Lightweight Directory Access Protocol (LDAP).[15]

2.3 Analysis Tools

The program tcpdump is used to examine the packets arriving at the various network interfaces.

3.0 Wireless Infrastructure

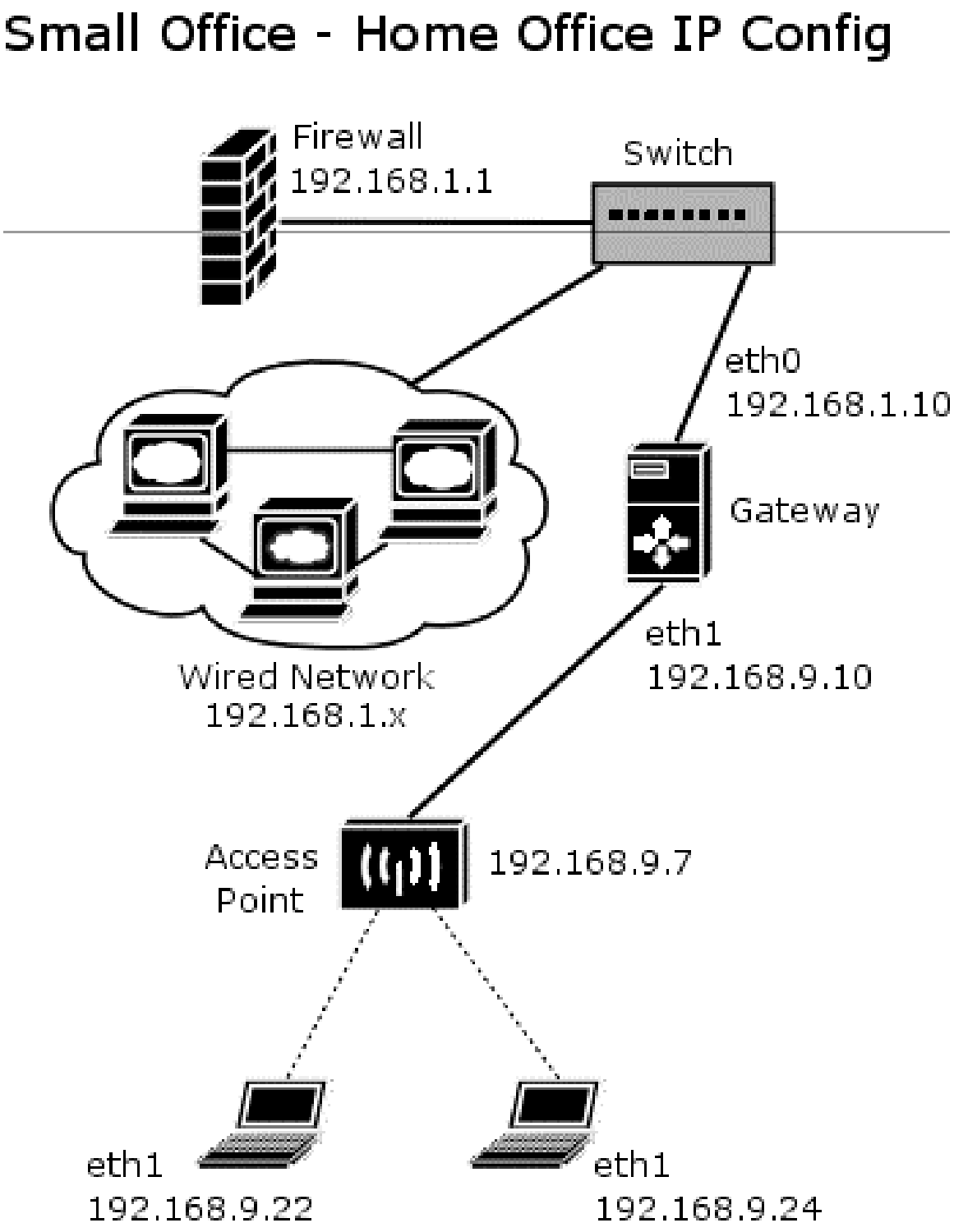
The wireless infrastructure is provided by the combination of a wireless gateway, hub, and wireless access point. Unless more than one access point is required, the hub is not required except that it does simplify the debugging of IP traffic between the wireless access point and the wireless gateway. Wireless clients, such as laptops running Gentoo Linux, connect to the network via the access

[14] "Upgrading to FreeS/WAN 2.x." http://www.freeswan.org/freeswan_trees/freeswan-2.00/doc/upgrading.html

[15] strongSec. http://www.strongsec.com/freeswan.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

point.



3.1 Gateway

The wireless gateway (gw.myorg.example.com) can be a legacy system, such as a Pentium 200 MMX PC, with two Ethernet adapters. One adapter is connected

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

to the wired portion of the network and the other to the hub. The primary function of this computer is to route valid traffic between the wireless clients and the other part of the network. All connections to the wireless gateway must be encrypted and authenticated. This reduces the likelihood that a cracker can view the traffic and/or break into the network.

At the time that this was written, there was no way to automatically include FreeS/WAN 2.0 in Gentoo without manually building and installing the system. Appendix A describes this process and configuration is described in the body of this paper.

3.2 Access Point

The wireless access point, for example, an Orinoco AP-200, bridges traffic from the wireless realm into the gateway. Due the need to provide "Defense in Depth", the maximum number of security features are specified (e.g., WEP, do not broadcast SSID, etc.) These represent the first line of defense against attacks. Theoretically, it is possible for someone to break into the wireless network by using a tool like AirSnort.

4. Gateway Configuration

Configuration of the gateway encompasses several attributes: routing IP traffic, FreeS/WAN server, and iptable filtering.

To enable the forwarding of packets from one Ethernet adapter to the other, /proc/sys/net/ipv4/ip_forward must be set to 1. If not, no traffic is routed.

FreeS/WAN server configuration requires several sets. Appendix B shows the mechanism used to create a X.509 certificate. Once they have been created, you must do the following steps:

1. Copy files to the appropriate directories underneath /etc/ipsec.d. Note that prior to FreeS/WAN 2.0, apparently the system (e.g. gw and ilaptop) .pem files existed in /etc/ipsec.d.

   ```
   cp cacert.pem /etc/ipsec.d/cacerts
   cp gw.myorg.example.org.pem /etc/ipsec.d/certs
   cp ilaptop.myorg.example.org.pem /etc/ipsec.d/certs
   cp crl.pem /etc/ipsec.d/crls
   cp gw.myorg.example.org.key /etc/ipsec.d/private
   ```

2. Edit the file /etc/ipsec.secrets so that it contains the PEM password specified when generating the certificate for gw:

   ```
   : RSA gw.myorg.example.org.key "Test...."
   ```

3. The full contents of /etc/ipsec.conf is shown in Appendix C. The client-specific information is shown below:

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
conn ilaptopClientNet
        leftsubnet=0.0.0.0/0
        also=ilaptopClient

conn ilaptopClient
        left=192.168.9.10
        leftcert=gw.myorg.example.org.pem
        rightcert=ilaptop.myorg.example.org.pem
        pfs=yes
        also=customConn
```

The IPtables below prevent unencrypted traffic from being transmitted over the wireless network and forward traffic between the wired network and the IPsec secured wireless network. All traffic is allowed to enter the gateway from the wired side of the network.

```
iptables –A FORWARD –i eth0 –o eth1 –j DROP
iptables –A FORWARD –i eth1 –o eth0 –j DROP
iptables –A FORWARD –i ipsec+ -o eth0 –j ACCEPT
iptables –A FORWARD –i eth0 –o ipsec+ -J ACCEPT
```

5. Linux Client Configuration

The network configuration of the Linux client utilizes a static IP address. All networking information must be entered into the configuration file. This provides an additional layer of security at the cost of convenience.

Because the configuration file format changed between FreeS/WAN versions 1.9x and 2.0, I did not confirm the interoperability of different versions of the software. I did the configuration using FreeS/WAN 2.0 with X.509 patches, built and installed as described in Appendix A.

FreeS/WAN client configuration requires several files. As indicated earlier, Appendix B shows the procedure used to create a X.509 certificate. Once they have been created, you must do the following steps:

1. Copy files to the appropriate directories underneath /etc/ipsec.d. Note that prior to FreeS/WAN 2.0, apparently the system (e.g. gw and ilaptop) .pem files existed in /etc/ipsec.d.

```
cp cacert.pem /etc/ipsec.d/cacerts
cp gw.myorg.example.org.pem /etc/ipsec.d/certs
cp ilaptop.myorg.example.org.pem /etc/ipsec.d/certs
cp crl.pem /etc/ipsec.d/crls
cp ilaptop.myorger.example,org.key /etc/ipsec.d/private
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

2. Edit the file /etc/ipsec.secrets so that it contains the PEM password specified when generating the certificate for gw:

   : RSA ilaptop.myorg.example.org.key “Test....”

3. The full contents of /etc/ipsec.conf is shown in Appendix D. However, the important parts are:

   ```
   conn ilaptopClientNet
           leftsubnet=0.0.0.0/0
           also=ilaptopClient

   conn ilaptopClient
           left=192.168.9.10
           leftcert=gw.myorg.example.org.pem
           rightcert=ilaptop.myorg.example.org.pem
           pfs=yes
           also=customConn
   ```

5. Verification and Debugging

5.1 Ready to Create IPsec Links

Issuing the command “ipsec verify” can be used to check the status of your FreeS/WAN installation and configuration. If you choose to use IPsec as a module, then you need to configure it to be loaded at boot time. If you don't, then ipsec verify will show that there is no “KLIPS support in kernel” (e.g., FAILED).

After a fresh install, you nay need to start “pluto” using the /etc/init.d/ipsec start shell script. This would be typically invoked when the box boots. The verify line is “Checking that pluto is running”.

5.2 Testing IPsec Links

The program tcpdump can be used to test whether or not the IP traffic is occurring over an IPsec link. In our configuration above, running tcpdump on the gateway:

```
gw # tcpdump –n –i eth1
```

Tcpdump must be run as superuser. The –n flag prevents conversion of DNS name to IP address and the –i flag identifies the desired interface.

After issuing this command, issues a command to generate traffic from the wireless client to the gateway or beyond. The output from tcpdump should show ESP packets, meaning that the data is encrypted. If the output does not have ESP, then it is not occurring over an IPsec link.

© SANS Institute 2003, Author retains full rights.

Connectivity between wireless clients must also be encrypted because the entire network connection is not trusted. One can perform a similar test where client one issues the ping while client two is using tcpdump to confirm that the traffic is not in the clear.

5.3 Debugging IPsec

To debug IPsec links, you need to check that the routing configuration is correct. Using the program "route –nv" or "ip route" reveal interesting information.

The command "ipsec barf" can be used to dump a great deal of information about the configuration of Linux, IP, and IPsec.

6. Ongoing Administration

6.1 Addition

When adding a new client to the wireless network, one must obtain the following information:

- MAC address of wireless adapter – if MAC address filtering is enabled on the access point, then the network administrator must program the user's new MAC address into the MAC filter control list. Optionally, the user could provide the wireless adapter so that the MAC address can be retrieved.
- X.509 information used for distinguished name – use whatever characteristics are important for your installation (e.g., country name, state name, city name, common name, and/or e-mail address).
- A pass phrase for use when generating the certification

The network administrator must use this information to generate a new certificate for the user, update the configuration files on the FreeS/WAN server, and configure the wireless client. If the wireless adapter was provided, then programming of the WEP key can be performed by the network administrator, retaining the confidentiality of the WEP key.

Given that the intended network is so small, it is likely that the administrator and users are located in the same location, greatly simplifying this.

6.2 Removal

To remove a client from the list of authorized users of the wireless network, one must remove the MAC address and configuration data from FreeS/WAN.

7. Conclusions

This document has provided a description on how FreeS/WAN can be used with X.509 certificates to secure a wireless network. This can be extended to include other untrusted network media. The network configuration assumes that the gateway is directly accessible to the clients and not hidden via NAT.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

8. References

Carlson, Nate. “Configuring an IPsec Tunnel between FreeS/WAN and Windows 2000/XP.” URL: http://www.natecarlson.com/linux/ipsec-x509.php.

Borisov, Nikita, Ian Goldberg, and David Wagner. “Intercepting Mobile Communications: The Insecurity of 802.11”. *7th Annual International Conference on Mobile Computing and Networking,* July 16-21, 2001. URL: http://www.isaac.cs.berkeley.edu/isaac/mobicom.pdf.

“Security of the WEP algorithm.” URL: http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html.

“Default Wireless Configurations.” URL: http://www.cirt.net/cgi-bin/ssids.pl.

Arar, Yardena. “Wi-Fi Networks Get a Security Upgrade.” PCWorld.COM. April 30, 2003. URL: http://www.pcworld.com/news/article/0,aid,110520,00.asp.

OpenSSH. URL: http://www.openssh.org

“The IPsec protocols.” URL: http://www.freeswan.org/freeswan_trees/2.0/ipsec.html.

Kent, S. and R. Atkinson. “IP Encapsulating Security Payload (ESP).”The Internet Engineering Task Force. Novermber 1998. URL: http://www.ietf.org/rfc/rfc2406.txt?number=2406.

Kent, S. and R. Atkinson. “IP Authentication Header.” The Internet Engineering Task Force. November 1998. URL: http://www.ietf.org/rfc/rfc2402.txt?number=2402.

Kent, S. and R. Atkinson. “The Internet IP Security Domain of Interpretation for ISAKMP.” The Internet Engineering Task Force. November 1998. URL: http://www.ietf.org/rfc/rfc2407.txt?number=2407.

Hughes, Woody. “Iptables, Linux and You!” GSEC Practical Assignment v.1.4b (Option 1). March 3, 2003. URL: http://www.giac.org/practical/GSEC/Woody_Hughes_GSEC.pdf

Portage. URL: http://www.gentoo.org/doc/en/faq.xml

“Upgrading to FreeS/WAN 2.x.” http://www.freeswan.org/freeswan_trees/freeswan-2.00/doc/upgrading.html

strongSec. http://www.strongsec.com/freeswan.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Appendix A: Building and Installing FreeS/WAN 2.0

1. Obtain a Linux kernel. I utilized the latest "stable" kernel – 2.4.21. I retrieved it from http://www.kernel.org or one of its mirror sites.
2. Untar the kernel into /usr/src/linux-2.4.21.
3. There are three options to consider:
   - "standard" FreeS/WAN – current version is 2.00
   - "standard" FreeS/WAN with user-supported patches such as X.509
   - an "unofficial" FreeS/WAN that has been prepatched.
4. For the purposes of this discussion, we will patch the standard FreeS/WAN to support X.509:
   - Obtain FreeS/WAN from ftp://ftp.xs4all.nl/pub/crypto/freeswan (used freeswan-2.00.tar.gz).
   - Obtain X.509 patch from http://www.strongsec.com/freeswan. This patch allows one to use RSA-based authentication via "X.509 or OpenPGP certificates between a Linux FreeS/WAN security gateway and an unlimited number of IPsec peers."
5. Untar the FreeS/WAN sources into /usr/src/freeswan-2.00.
6. Untar the X.509 patch into /usr/src/x509-1.3.5-freeswan-2.00
7. Apply patch by entering the directory /usr/src/freeswan-2.00 and issuing the command: patch –p1 < /usr/src/x509-1.3.5-freeswan-2.00/freeswan.diff
8. Edit freeswan-2.00/programs/pluto/Makefile by removing comment to enable LDAP_URL.
9. Make sure there is a link from /usr/src/linux to the Linux kernel source tree – this is used by the FreeS/WAN patch mechanism.
10. Configure the Linux kernel using menuconfig, oldconfig, xconfig (or other means).
11. Patch the kernel by issuing: make precheck verset kpatch in the freeswan directory:
    - precheck –
    - verset –
    - kpatch – does the kernel patch
12. Build the IPsec patched Linux kernel via: make clean dep bzImage modules modules_install.
13. Build FreeS/WAN by issuing make confcheck programs install.

Example:

Retrieve latest stable kernel:

```
cd /tmp
wget ftp://www.kernel.org/pub/linux/kernel/v2.4/linux-2.4.21.tar.gz
wget ftp://www.kernel.org/pub/linux/kernel/v2.4/linux-2.4.21.tar.gz.sign
gpg –verify linux-2.4.21.tar.gz.sign linux-2.4.21.tar.gz
tar –xzf linux-2.4.21.tar.gz –C /usr/src
rm linux-2.4.21.tar.gz linux-2.4.21.tar.gz.sign
cd /usr/src
mv linux linux.old
ln –s linux-2.4.21 linux
```

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

Retrieve latest FreeS/WAN and apply X.509 patch:

```
cd /tmp
wget ftp://ftp.xs4all.nl/pub/crypto/freeswan/freeswan-2.00.tar.gz
wget ftp://ftp.xs4all.nl/pub/crypto/freeswan/freeswan-2.00.tar.gz.md5
md5sum -c freeswan-2.00.tar.gz.md5
freeswan-2.00.tar.gz: OK
tar -xzf freeswan-2.00.tar.gz -C /usr/src
rm freeswan-2.00.tar.gz freeswan-2.00.tar.gz.md5

wget http://www.strongsec.com/freeswan/x509-1.3.5-freeswan-2.00.tar.gz
wget http://www.strongsec.com/freeswan/x509-1.3.5-freeswan-2.00.tar.gz.md5
md5sum -c x509-1.3.5-freeswan-2.00.tar.gz.md5
x509-1.3.5-freeswan-2.00.tar.gz: OK
tar -xzf x509-1.3.5-freeswan-2.00.tar.gz -C /usr/src
rm x509-1.3.5-freeswan-2.00.tar.gz x509-1.3.5-freeswan-2.00.tar.gz.md5

cd /usr/src/freeswan-2.00
patch -p1 < ../x509-1.3.5-freeswan-2.00/freeswan.diff
```

Kernel Configuration:
cd /usr/src/linux
vi Makefile – to update version number, e.g. Add "-Ipsec" to EXTRAVERSION".
Make menuconfig

cd ../freeswan-2.00
make precheck verset kpatch
cd ../linux
make menuconfig – needed to generate new autoconf.h file. If you are building a kernel include Ipsec without using modules, you need to disable and then re-enable CONFIG_IPSEC; otherwise, eventhough it indicates "y", it is actually a "M".

Building Patched Kernel:

```
cd ../linux
make clean dep
make bzImage modules modules_install
```

Building User-Mode FreeS/WAN Programs:

```
cd ../freeswan-2.00
make confcheck programs install
reboot
```

© SANS Institute 2003, author retains full rights.

Appendix B: Creating an X.509 Certificate Authority (CA)[16]

The process for creating an X.509 Certificate Authority (CA) is illustrated below. This process must be performed once, uses OpenSSL, and was based on the procedure described by Nate Carlson on his Website.

```
# /etc/ssl/misc/CA.pl -newca
CA certificate filename (or enter to create)

Making CA certificate ...
Using configuration from /etc/ssl/openssl.cnf
Generating a 2048 bit RSA private key
...........+++
.........................+++
writing new private key to './demoCA/private/cakey.pem'
Enter PEM pass phrase:
Verifying password - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [US]:US
State or Province Name (full name) [Some-State]:MyState
Locality Name (eg, city) []:MyCity
Organization Name (eg, company) [Internet Widgits Pty Ltd]:My Organization
Organizational Unit Name (eg, section) []:
Common Name (eg, YOUR name) []:CA
Email Address []:ca@myorg.example.org
```

© SANS Institute 2003, Author retains full rights.

[16] Carlson, Nate. "Configuring an IPsec Tunnel between FreeS/WAN and Windows 2000/XP." http://www.natecarlson.com/linux/ipsec-x509.php.

Appendix C: Generating X.509 Certificates[17]

This procedure is used to generate certificates for each machine utilizing FreeS/WAN and was derived from content on Nate Carlson's Website. The CA.pl perl script is invoked two times: once to generate the certificate and a second time to sign it. Then the files are renamed to more appropriate names and the key file is editted to contain the appropriate content.

```
# /etc/ssl/misc/CA.pl -newreq
Using configuration from /etc/ssl/openssl.cnf
Generating a 2048 bit RSA private key
.....................................................................+++
.......................................+++
writing new private key to 'newreq.pem'
Enter PEM pass phrase:
Verifying password - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [US]:
State or Province Name (full name) [Some-State]:MyState
Locality Name (eg, city) []:MyCity
Organization Name (eg, company) [Internet Widgits Pty Ltd]:My Organization
Organizational Unit Name (eg, section) []:
Common Name (eg, YOUR name) []:gw.myorg.example.org
Email Address []:admin@myorg.example.org

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
Request (and private key) is in newreq.pem

# /etc/ssl/misc/CA.pl -sign
Using configuration from /etc/ssl/openssl.cnf
Enter PEM pass phrase:
Check that the request matches the signature
Signature ok
The Subjects Distinguished Name is as follows
countryName           :PRINTABLE:'US'
stateOrProvinceName   :PRINTABLE:'MyState'
localityName          :PRINTABLE:'MyCity'
organizationName      :PRINTABLE:'My Organization'
commonName            :PRINTABLE:'gw.myorg.example.org'
emailAddress          :IA5STRING:'admin@myorg.example.org'
Certificate is to be certified until Jun 25 13:33:03 2005 GMT (730 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

[17] Carlson, Nate. "Configuring an IPsec Tunnel between FreeS/WAN and Windows 2000/XP." http://www.natecarlson.com/linux/ipsec-x509.php.

© SANS Institute 2003, As part of GIAC practical repository. Author retains full rights.

```
Signed certificate is in newcert.pem

# mv newcert.pem gw.myorg.example.org.pem
# mv newreq.pem gw.myorg.example.org.key
```

The final step is to edit the key file – deleting content until the file starts with "BEGIN RSA PRIVATE KEY" and ends with "END RSA PRIVATE KEY". Essentially, you are removing the section that starts with "BEGIN CERTIFICATE REQUEST". These files are now ready to be copied over to the appropriate systems.

This process should be repeated for each system utilizing FreeS/WAN.

© SANS Institute 2003, Author retains full rights.

Appendix D: /etc/ipsec.conf for FreeS/WAN Server

```
# /etc/ipsec.conf - FreeS/WAN IPsec configuration file

version      2.0    # conforms to second version of ipsec.conf specification

# basic configuration
config setup
       interfaces="ipsec0=eth1"
       # Debug-logging controls:  "none" for (almost) none, "all" for lots.
       # klipsdebug=all
       # plutodebug=dns
       uniqueids=yes

conn %default
       keyingtries=1
       compress=yes
       disablearrivalcheck=no
       authby=rsasig
       leftrsasigkey=%cert
       rightrsasigkey=%cert

# include common information regarding each client
# this information is the same for both clients and the server
include ipsec.d/conns/*.conf

# this customConn information contains the information that is unique
# for servers
conn customConn
       right=%any
       auto=add

# disable Opportunistic Encryption - configured by default
conn block
       auto=ignore

conn private
       auto=ignore

conn private-or-clear
       auto=ignore

conn clear-or-private
       auto=ignore

conn clear
       auto=ignore

conn packetdefault
       auto=ignore
```

© SANS Institute 2003, Author retains full rights.

Appendix E: /etc/ipsec.conf for FreeS/WAN Client

```
# /etc/ipsec.conf - FreeS/WAN IPsec configuration file

version      2.0    # conforms to second version of ipsec.conf specification

# basic configuration
config setup
       interfaces=%defaultroute
       # Debug-logging controls:  "none" for (almost) none, "all" for lots.
       # klipsdebug=all
       # plutodebug=dns
       uniqueids=yes

conn %default
       keyingtries=1
       compress=yes
       disablearrivalcheck=no
       authby=rsasig
       leftrsasigkey=%cert
       rightrsasigkey=%cert

# include common information regarding each client
# this information is the same for both clients and the server
include ipsec.d/conns/*.conf

# this customConn information contains the information that is unique
# for clients
conn customConn
       right=%defaultroute
       auto=start

# disable Opportunistic Encryption - configured by default
conn block
       auto=ignore

conn private
       auto=ignore

conn private-or-clear
       auto=ignore

conn clear-or-private
       auto=ignore

conn clear
       auto=ignore

conn packetdefault
       auto=ignore
```

© SANS Institute 2003, Author retains full rights.