Interested in learning
more about security?

SANS Institute
InfoSec Reading Room

This paper is from the SANS Institute Reading Room site. Reposting is not permitted without express written permission.

# Pocket Nessus

The purpose of this paper is to introduce a particular set of tools that I've found work well for my applications. I, too, recommend Nessus along with another useful tool - the White Glove CD - which contains many other applications that can be used to secure your network. In the end, you will find that you have an extremely portable toolbox that fits in your pocket and runs an open source vulnerability scanner recommended by networking organizations, SANS instructors, and used by many commercial companies. This paper ...

Copyright SANS Institute
Author Retains Full Rights

AD
CounterTack
Counter Tack Active Forensics for In-Progress Attacks
GET THE WHITE PAPER NOW >>>

Pocket Nessus
By Tony Enriquez
Assignment version: 1.2f

**Introduction**
Choosing the right set of security tools for your network can be a daunting task. Do you select the most popular tools recommended by professionals in the field? Do you pay for the applications or use the ones that cost nothing? Do you use tools from different sources or find many tools consolidated into one package? The answers to those questions are numerous but often the best answers come from those who have actually used the tools successfully and found them useful.

While in pursuit of the right set of tools, you will find one program that is frequently mentioned – Nessus. If you are putting together a network security toolbox, many people will tell you that Nessus is one of those "must have tools." According to John Green, author and SANS instructor, "Any set of tools should, at a minimum provide the following three capabilities: network mapping (Nmap), passive vulnerability assessment (Tcpdump), active penetration testing (Nessus)." Once you decide on which tools to use, you need a platform to run them. You could load up your tools onto a laptop, but then you would have to bring your laptop, which may prove to be cumbersome. Wouldn't it be easier to leave your PC behind and run Nessus on one of the computers on the network under inspection? Those of you familiar with Nessus know that the server usually runs on a Unix type platform. What if all the computers are Windows based machines? Most of the computers I'm responsible for are x86 type processors, usually running a Microsoft OS, with a few users running Linux. I could install Nessus on one of the Linux boxes, but I don't think the users of those machines would want me to put any software on to their computers. There are several solutions for this problem. One solution would be to boot into another OS from CDROM or floppy and run entirely from RAM.

The purpose of this paper is to introduce a particular set of tools that I've found work well for my applications. I, too, recommend Nessus along with another useful tool – the White Glove CD – which contains many other applications that can be used to secure your network. In the end, you will find that you have an extremely portable toolbox that fits in your pocket and runs an open source vulnerability scanner recommended by networking organizations, SANS instructors, and used by many commercial companies. This paper will provide a brief introduction to Nessus and the White Glove CD. I will then explain how I use the White Glove CD to run Nessus.

**What is Nessus?**
Renaud Deraison, Nessus developer, states on the Nessus.org website that "The "Nessus" Project aims to provide the Internet community a free, powerful, up-to-date and easy to use remote security scanner." According to the introduction on the Nessus.org website, "The security scanner is a software which will audit remotely a given network and determine whether bad guys (aka 'crackers') may break into it, or misuse it in some way" (Deraison).

**How does Nessus work?**
Nessus is comprised of a server – client architecture. The server does the work of finding the holes and reporting them back to the client. The client provides the GUI to the user and does the

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

work of displaying the results found by the server counterpart. As stated on the Nessus FAQ, "Nessus tries to find security problems by connecting to it's targets over IP and simulating various application protocols. When testing a problem in a web server for example, Nessus pretends to be a web browser (talking http). When testing a Windows fileserver, Nessus pretends to be a Windows client (talking SMB). Most of these tests are performed by (NASL) scripts. Each script is crafted to perform one specific task" (van den Berg and van der Kooij).

**Why use Nessus?**

A side-by-side comparison of vulnerability scanners conducted by Jeff Forristal and Greg Shipley in Network Computing Magazine, demonstrates Nessus' power and how it is just as effective as commercial scanners. According to the authors, "Some of us were a bit skeptical of the open-source Nessus project's thoroughness until it discovered the greatest number of vulnerabilities" and went on to say that Nessus got the best overall score because it did more things right than any other scanners they tested (Forristal and Shipley).

The high praise continues in the industry as Network Computing gave Nessus the winning Vulnerability Assessment Tool award in May 2001. "Why? … Because it works. It doesn't crash; it doesn't have an unusable interface; it doesn't hide vulnerability checks under the denial of service section; and it's being worked on actively by the security community at large," says the award judges (Fratto).

In fact, ease of use and power of Nessus is evident by the fact that commercial companies are also using Nessus to safeguard their networks. Just look at the following headline:

> "MELVILLE, N.Y., Feb. 26 -- VIGILANTe today announced the integration of the leading open-source security scanner, Nessus, into its premier automated Internet security assessment service, SecureScan. This addition to SecureScan bolsters an arsenal of commercially available, open- source and proprietary assessment tools" (VIGILANTe).

An extremely important reason to use Nessus is that it provides you with the ability to update your attack signatures. Nessus allows you to write your own attack scripts via NASL (Nessus Attack Scripting Language) or download "plugins" from the Internet. "Plugins" are the security scripts that run the "tests." If an exploit is released, chances are that someone in the security community will create an attack script, or "plugin," for Nessus. A real-world example can be seen in a recent advisory concerning the Windows Universal Plug and Play exploit (refer to www.cert.org/advisories/CA-2001-37.html). The exploit was released by CERT on 12/20/01. On 12/29/01 a Nessus upnp attack script was announced on the Nessus website (www.nessus.org/nasl/upnp_xp.nasl). I'm not implying that it's the fastest resource available, but it is extremely important to have access to - and run - an attack script if you don't have the resources or ability to create your own attack scripts. The ability to create or update the plugins is one of the greatest strengths of the Nessus application.

You can also use Nessus to gain auditing experience with your system/network environment. By using Nessus, the people responsible for securing your networks will gain a better understanding of their network. They will also become familiar with vulnerability scanners to help them evaluate other scanners. Sometimes a demo version will only let you scan one computer. Since Nessus does not limit you to what can be scanned, the administrator will have a better idea of

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

how long an entire network scan might take. Once they get a feel for how long their entire network will take, they can experiment with segmenting the scans for time specific operations. You always need a baseline to compare to and Nessus can provide you with that knowledge baseline. Once you scan your network with Nessus, you can fix/update all the vulnerabilities it finds. Rescan to verify, and then use those results as a reference for comparison scans made in the future.

Please note, I will not be explaining how to use Nessus in this paper. For specific instructions, refer to the links at the end of the paper.

**What is the White Glove CD?**
The White Glove CD, developed by Fred Cohen and Associates. According to the website:

> "The White Glove is our newest innovation in low-cost, flexible, portable system tools. The White Glove provides a bootable mini-CD-ROM complete with help, software, and remote access capabilities. It includes a nominal menu-based interface as well as full Linux command line capabilities, and includes remote access to White Glove servers on the Internet."

For this paper, I'll be using White Glove version 2.9.7, which is a miniature Linux distribution that runs on x86 machines. Please note: I've had success with computers as slow as Pentium 133s to dual 800 MHz PIIIs. When your system is completed booting from the CD, you now have a Linux platform to audit your network. The benefits? You can work from the command line or run X-windows for the GUI inclined. The tools on the CD? -- To list all of the tools would take a few pages, so I'll group some of the network analysis tools according to the capabilities John Green noted above.

**Network Mapping:** nmap, nmapfe, arping, hping, icmpenum, xprobe
**Passive Vulnerability Assessment:** tcpdump, ethereal, sniffit, ettercap, dsniff
**Active Penetration Testing**: Nessus, whisker

Beyond the network analysis function, this CD can be used as a forensic tool (tct – The Coroner's Toolkit and diskwipe), a firewall (iptables) and a server (sshd, http, nfs, smb, etc.). (Refer to http://all.net to see a complete list of features. Note that not all of the features listed at the all.net website are implemented in 2.9.7.)

Again, this is only a small list, and is current for the 2.9.7 version of White Glove. For more information about the different tools, refer to the links at the end of the paper.

Of course there are other bootable CDs, and some of them are free. For example, Trinux is another alternative. Although more familiar for it's floppy disk distribution, you can also use Trinux from a CD. See the links at the end of the paper for more information. However, the few that I have used just didn't have all the network analysis tools that the White Glove CD had on it.

**Using the White Glove CD**
Before we get started: The author assumes you have authorization to utilize the tools mentioned in the paper. Using the mentioned tools and other applications on the CD can easily bring a functioning network to a complete stop or accidentally erase an entire hard drive. Of course, the best way to learn is to actually practice the task at hand. I recommend a test network with various computers and network devices. You could also run the various tools (not White Glove) from one of the virtual-machine software packages (vmware, win4lin). You can run a virtual

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

computer and analyze the host computer, or vice versa. The important point here is to have permission first, in writing.

The author assumes the reader is familiar with Linux and Nessus. If additional information is required, refer to the links at the end of the paper. Please note: The version of Nessus on the White Glove CD is 1.1.8.

**Booting White Glove**

In order to boot the CD, the computer must be configured to boot from the CD. You can accomplish this by modifying settings in the computer's bios. Consult the computer's user manual or go the manufacturer's web site for specific bios information. Although most modern CD drives accept the 180 MB CD, I did run into a few machines that could not boot the smaller diameter CD. To remedy this problem, have the White Glove CD burned to a standard size (640MB) CD. Once you modify the bios to boot from the CD first, exit from the bios menu to reboot the computer. The following is taken directly from the White Glove tutorial called "Booting Up" on the Fred Cohen and Associates website (http://all.net) and describes the different modes of operation:

> "On bootup, the White Glove will go through its normal startup routine. The first thing you will see after system self-tests is a menu that looks something like this:
>
> ```
> linux          diskless          debug
> linux-800      diskless-800      memtest
> linux-1024     diskless-1024
> linux-1280     diskless-1280
> linux-1600     diskless-1600
> ```
>
> The bootup modes are as follows:
>
> - linux-xxx[x] boots from the CD mounting the CD's contents with the screen resolution for normal login screens set at the resolution selected by the number. The numbers indicate 800x600, 1024x768, 1280x1024, and 1600x1200 respectively. If the resolution you select does not work you will be prompted for alternatives.
> - diskless-xxx[x] boots from the CD but instead of mounting the CD as a device, it copies the CD contents into RAM and ejects the CD. In diskless mode you can then load additional CDs for use, or you can simply start the computer working for you and put the CD back in your shirt pocket.
> - debug mode puts you in a mode that provides absolutely the minimal functionality that can be created under Linux. There is really nothing you can do in this mode unless you are a White Glove wizard, but you can feel free to try.
> - memtest performs memory testing on your computer. It is a very handy memory diagnostic that we have used to debug memory problems on lots of computers. It will also tell you how much memory you have.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

- If you know how lilo and similar bootup systems work you can put in specific parameters for things like screen resolution, and so forth. You are entirely on your own for this.

You will have from 3 to 5 seconds from the appearance of this menu to make a selection. If no selection is made, it will default to 'linux' which is 80x25' text mode. After this, Linux will proceed to boot up. This typically takes from 10 to 45 seconds depending on details of your hardware. You will see a number of messages, including some things that may look like errors. Please enjoy reading them if you like. In the end, WG will either fail or succeed, leaving you with a situation where the computer will do nothing else (i.e., fail) or where you are prompted for login (i.e., succeed). If you are prompted for login, you are done booting up. Otherwise, you need to try another bootup selection (if you tried one in the first place)."

Diskless Mode Notes: When in diskless mode, not all of the binaries are copied into RAM. Therefore, if you're thinking that the entire CD's content is copied into RAM that is not the case. If you want to have all the applications available, I recommend that you choose one of the "Linux" modes.

**Configuration-Network**

Once the computer is up and running, the next step is to configure the network interfaces. The CD does a great job at automatically configuring most peripherals. If the module for your network interface card is not on the distribution, simply compile it on another computer (Kernel 2.4.12) and include it on a floppy. To load the desired module, mount the floppy that contains your module and run the "insmod" command.

```
# insmod /mnt/floppy/<module name>
```

Insert your module for the module name. Most modern computers, to include laptops, seem to boot up fine. The platform in use for this paper is a Dell Inspiron 700 PII, 128 MB of RAM, and a 3com 3CC575CT pcmcia network interface card.

Configuring network: So how can you tell if your nic is recognized? You can display the /proc/modules file to see if the appropriate module was loaded. You can also run the "ifconfig" command. If you see an entry for eth0, the OS recognizes your nic. FYI, the default IP address is 10.0.0.8 with a netmask for a class C. If the network address is known, use the "ifconfig" command to assign an address. The following command sets the eth0 interface IP address as 192.168.0.5 with a class C netmask, and activates it.

```
# ifconfig eth0 192.168.0.5 netmask 255.255.255.0 up
```

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

Configure a default gateway, if applicable.

```
# route add default gw 192.168.0.1
```

If you didn't know the network addresses, you can use one of the passive/sniffer applications (tcpdump, ethereal) and get a view of the network IP addresses. After analyzing the traffic on the network, you can make an educated guess for the network address and set your nic appropriately. Remember; do not to use an address already taken.

If you're on a DHCP (Dynamic Host Configuration Protocol) network, execute the "pump" command.

```
# pump
```

For more help on "pump," or any other common command, type help <command> from the command prompt. Not all of the commands have help pages, but most of them do. You can do a directory listing of the /usr/man/ to list all the help pages. Once the network interface is set correctly, you can test connectivity by pinging known hosts on your network.

Configuration-X11

```
# mouseconfig
```

This command runs the mouse configuration program. Just follow the prompts, select the appropriate option for your mouse device, and then select "Yes" to make the changes. If you don't run the "mouseconfig" command, your mouse will not work properly when you start the XFree86 GUI. The resolution of your X-windows environment depends on what mode you selected during bootup, your video adapter, and monitor. I can't offer any troubleshooting tips since I have not experienced trouble getting X-windows working. The oldest video card I had was a Number 9 332 and had no problems. For more information go to http://all.net/WG/tutor/X11.html and http://all.net/WG/tutor/XStuff.html. To start the GUI, use the "startx" command:

```
# startx
```

The GUI will start up and may look like the screen in Figure 1:

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

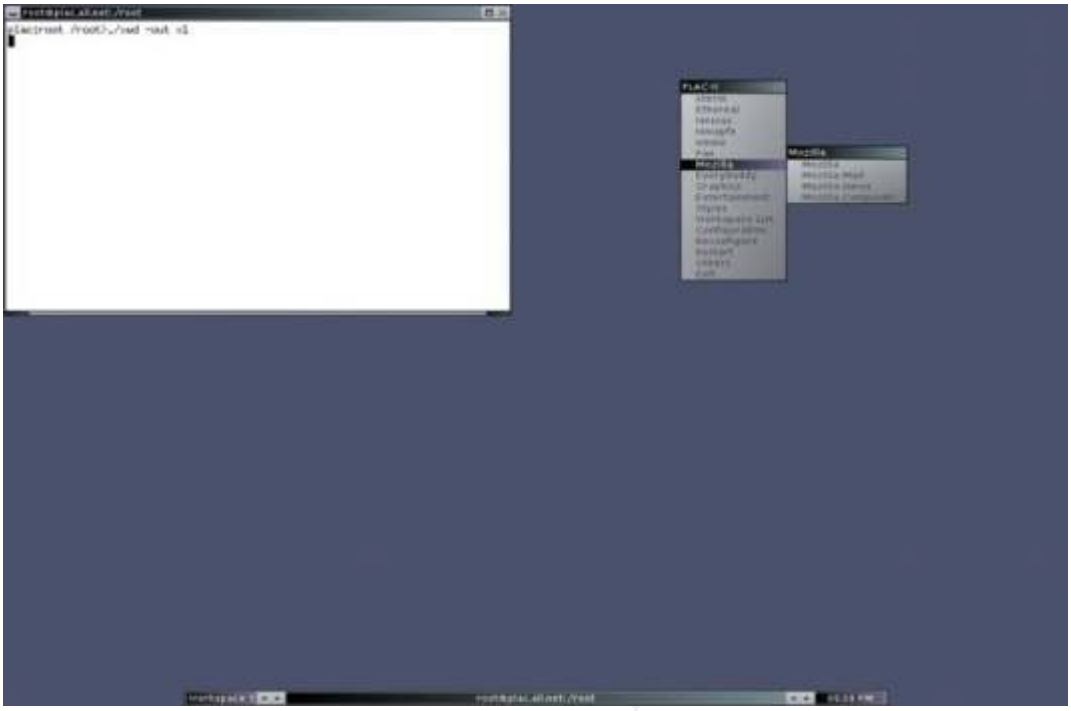

Figure 1

**Configuration-Nessus**

The author assumes you are familiar with Linux and Nessus. If you have specific questions on how Nessus works, refer to the links at the end of the paper. The version of Nessus on my White Glove CD is 1.1.8. This is a development version of Nessus, but I have yet to run into major problems.

**<u>Step 1 – Modify nessusd.conf file</u>**

In order to run the initial setup scripts, you have to modify the nessusd.conf file. This configuration file tells the nessusd server where information is retrieved and written. The default setup on this version of White Glove has default directories that are on the CD. If the server has to "write" to the CD, the server will crash and you'll never run Nessus. Instead of going through every line on the nessusd.conf file, I'll highlight the entries that must be changed. What directory you change it to doesn't really matter, as long as the directory you choose is in RAM.

```
# Log file (or 'syslog') :
logfile = /root/nessusd.messages
```

```
# Dump file for debugging output, use `-' for stdout
dumpfile = /root/nessusd.dump
```

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

```
# Rules file :
rules = /etc/Nessus/nessusd.rules

# Users database :
users = /etc/Nessus/nessusd.users



# Crypto options :
negot_timeout = 600
peks_username = nessusd
peks_keylen = 1024
peks_keyfile = /etc/nessus/nessusd.private-keys
```

As I mentioned earlier, the ability to update the plugins is one of the greatest strengths of the Nessus application. In order to update the plugins, via the Internet or from a storage device, you have to configure the nessusd.conf file to indicate where the plugins are stored. Since the default location is on the read-only CD, you can't add new plugins. All I did was modify the location of the plugins folder from the default location, to a location in RAM. Then I copied the plugins folder from the CD to the directory indicated in the nessusd.conf file. As of December 29, 2001, there were 811 plugins available from the http://www.nessus.org website that take up approximately 12.5 MB. The following is the entry that I modified in the nessusd.conf file.

```
# Path to the security checks folder:

      plugins_folder = /etc/nessus/plugins
```

Then I copied the plugins folder to that directory:

```
# cp -r /CD/sectools/lib/nessus/plugins /etc/nessus/
```

**Step 2 – nessus-mkcert**

This command configures the SSL certificate. The "development" version of Nessus utilizes SSL certificates for additional authentication and security. I don't use this Nessus setup with a remote client, (I run the GUI client locally) so I usually use the default entries/answers when setting up the certificate. Since the OS is kept in RAM, all the certificate info is gone when you power off the computer.

```
# nessus-mkcert
-----------------------------------------------------------------------------
                    Creation of the Nessus SSL Certificate
-----------------------------------------------------------------------------

This script will now ask you the relevant information to create the SSL
certificate of Nessus. Note that this information will *NOT* be sent to
anybody (everything stays local), but anyone with the ability to connect to
your Nessus daemon will be able to retrieve this information.
```

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

```
CA certificate life time in days [1460]:
Server certificate life time in days [365]:
Your country (two letter code) [US]:
Your state or province name [none]:
Your location (e.g. town) [Paris]:
Your organization [Nessus Users United]:

-------------------------------------------------------------------------------

Congratulations. Your server certificate was properly created.

/usr/local/etc/nessus/nessusd.conf updated

The following files were created :

. Certification authority :
   Certificate = /usr/local/com/nessus/CA/cacert.pem
   Private key = /usr/local/var/nessus/CA/cakey.pem

. Nessus Server :
    Certificate = /usr/local/com/nessus/CA/servercert.pem
    Private key = /usr/local/var/nessus/CA/serverkey.pem

Press [ENTER] to exit
```

The server will not run until you set up the SSL certificate. Now that the SSL is configured, the next step is to add a user.

**Step 3 – nessus-adduser**

The "nessus-adduser" command configures who can access the Nessus server. The only user on the system I use is root. You can setup the White Glove CD to have multiple users (it's a Linux box!) but for simplicity, the example below shows how I setup Nessus for user "root," with a password of 123123. Of course, this password is an extremely poor choice. The learning point here is that when you type in the password, it's shown on the screen. Keep that in mind if you have people looking over your shoulder. Here are some of the variables:

Login: <user you want to allow access to Nessus server>
Authentication: pass = password cert = SSL certificate
User Rules: Allows you to restrict certain networks/nodes to specific users.
This is what it looks like when you execute this command:

```
# nessus-adduser
Using /var/tmp as a temporary file holder

Add a new nessusd user
----------------------
Login : root
Authentication (pass/cert) [pass] :
Login password : 123123

User rules
```

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

```
----------
Nessusd has a rules system which allows you to restrict the hosts
that root has the right to test. For instance, you may want
him to be able to scan his own host only.

Please see the nessus-adduser(8) man page for the rules syntax

Enter the rules for this user, and hit ctrl-D once you are done:
(the user can have an empty rules set)

Login             : root
Password          : 123123
DN                :
Rules             :

Is that ok ? (y/n) [y]
user added.
#
```

## Step 4 – Starting Nessusd

The simplest command to start the server is:

```
# nessusd -D
```

This command will start the server on it's default port of 1281. If you wanted to run this server on a different port, you can add a –p <port#> to the command above.

```
# nessusd -D -p 35000
```

That should do it. If you receive any errors, chances are a line from the nessusd.conf file points to a location on the CD and can't write to it. If you don't receive any errors, the server has probably started. You can confirm this by starting the Nessus client. The White Glove has a shortcut to it by right clicking the mouse and selecting "Nessus." Enter the appropriate port number, user and password. Leave the Nessusd Host as localhost. If you are successful, the display will look like Figure 2.

© SANS Institute 2002, Author retains full rights.

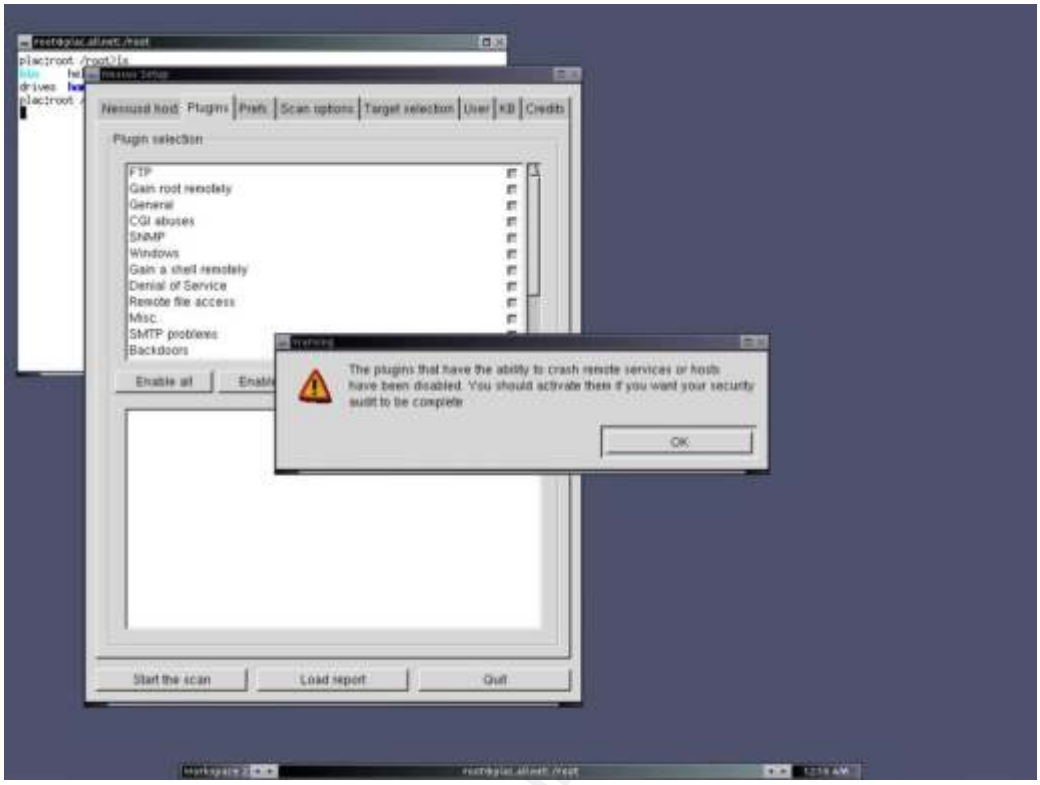


Figure 2

From here you can begin to assess your network. Please note: I won't cover the "How-To-Use Nessus" in this paper. However, I will provide an example of how Nessus can be used. The target was a "Default Windows XP" box, heavy emphasis on Default. The following screen shots will display two security holes. Figure 3 is the Universal Plug and Play vulnerability and Figure 4 indicates the "Null Session" vulnerability.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

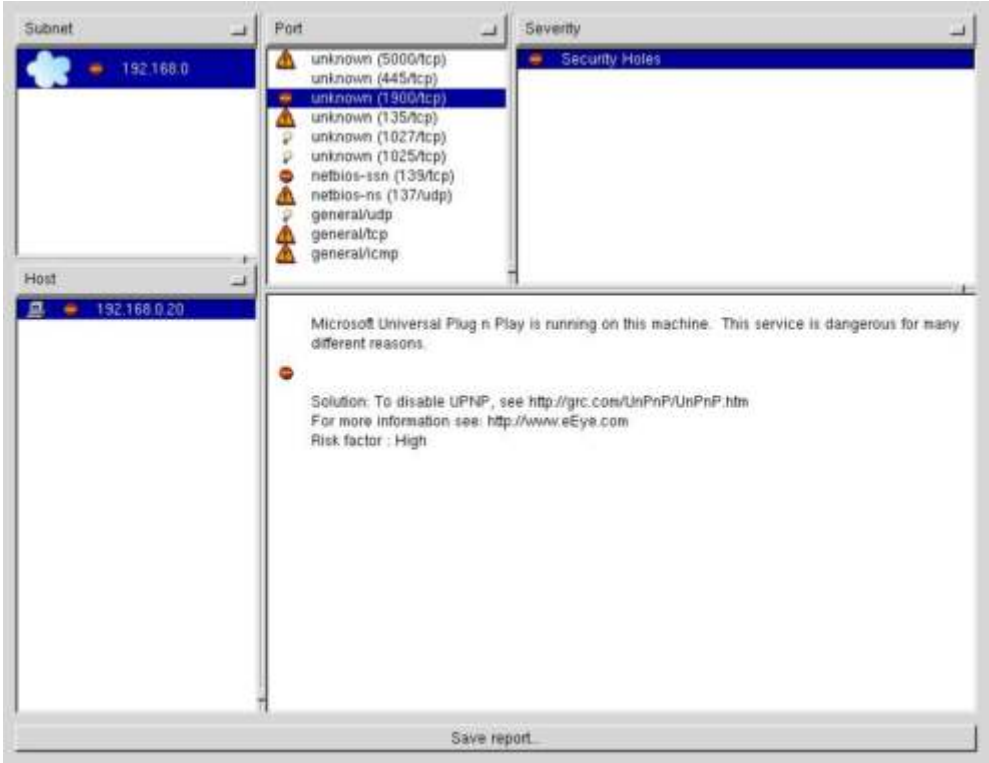


Figure 3

Figure 4

**Updating Attack Scripts**

If I didn't update the "plugins," Nessus would have found port 1900 open, but not report the

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

UPNP vulnerability. Nessus comes with an update script that gets the latest plugins and installs them onto your computer. Depending on who you talk to, this is either a great idea, or a terrible idea. The best that can happen is you run the script and you receive the latest and greatest of the Nessus plugins. The worst that can happen is – well whatever you can dream up – because someone could probably make it happen. An example could be that someone tricked you into going to another site and you downloaded the latest Trojan that appears as a Nessus plugin. You can always download the plugins from the www.nessus.org website and go through each plugin by verifying the attack script, or update them from your own trusted site.

In order for you to use the update script, you need to modify the nessus-update-plugins script. Specifically, you need to change the path for "wget."

```
fetch_cmd="/usr/bin/wget -q -O -"
```

to

```
fetch_cmd="/CD/local/bin/wget -q -O -"
```

Then I modified the following lines so the script would place the updated plugins scripts into my /etc/nessus/plugins directory in RAM.

```
prefix=/usr/tools
```

to

```
prefix=/etc
```

and

```
libdir=${exec_prefix}/lib
```

to

```
libdir=${exec_prefix}
```

There is always another way to do something, but this modification of the Nessus-update plugins file worked for me.

I know what some of you are thinking - these are a number of modifications to make each time I boot up a White Glove CD. However, Fred Cohen and Associates have already thought about settings that a user might want to have as a default. The solution is the plac.go file. This file contains all the commands that you want to execute to help you "pre-configure" your soon-to-be Linux box. The plac.go file is stored on a DOS formatted floppy and is read during boot up. Once the system is up and running, before the login prompt, the commands in the plac.go file are executed. Here is a listing of the floppy and the contents of my plac.go file for this particular project:

```
plac:root /mnt/floppy>ls -al
total 45
drwxr-xr-x   2 root     root         7168 Jan  1  1970 .
drwxr-xr-x   6 root     x               0 Jan  8 17:24 ..
-rwxr-xr-x   1 root     root         4888 Dec 28 18:42 nessus-update-plugins
-rwxr-xr-x   1 root     root         3103 Dec 28 17:28 nessusd.conf
-rwxr-xr-x   1 root     root         3118 Dec 28 14:30 nessusd.conf.old
```

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

```
-rwxr-xr-x   1 root     root          436 Jan  3 08:56 plac.go
-rwxr-xr-x   1 root     root          434 Dec 28 14:30 resolv.conf
-rwxr-xr-x   1 root     root        20964 Jan  3 14:15 xwd
```

plac.go file:

```
# load modules
# insmod /mnt/floppy/<module name>
#
# copy my resolv.conf file to /etc
cp /mnt/floppy/resolv.conf /etc/

#mkdir -p /etc/nessus/

cp /mnt/floppy/nessusd.conf /etc/nessus/
cp /mnt/floppy/nessus-update-plugins /etc/nessus/
cp -r /CD/sectools/lib/nessus/plugins_factory /etc/nessus/
cp -r /CD/sectools/lib/nessus/plugins /etc/nessus/
ifconfig eth0 192.168.0.5 netmask 255.255.255.0 up
route add default gw 192.168.0.1
```

The "xwd" program is the application I used to take the screen shots from the White Glove CD.

**It's just a tool**

Nessus only looks for vulnerabilities it knows about. If you are using Nessus to identify specific exploits, be sure to read the actual <exploit>.nasl file for plugin dependencies. Some plugins require information generated from other plugins before they will run. As an example, we loaded up a computer with the SubSeven Trojan running on port 27374. Nessus could identify that port 27374 was open but needed input from the Queso plugin before it would run the SubSeven plugin. Be sure to enable those required plugins. As with anything else, it takes the user's cranial housing group to make sense of all the data that is generated from Nessus. Take the time to research each exploit. If you have the skill set available, run the actual exploit on a test machine, then scan it with Nessus to verify your custom plugins, or plugins available from the web site.

For future reference, you can save your output as a .nsr file that can be read with the Nessus Client program. Another convenient thing to do is save the output as html, open it in Mozilla (it's included on the White Glove CD) and print it via a network printer.

**Conclusion**

By using the White Glove CD as a platform to run Nessus you can audit your network devices from nearly any x86 platform on the network. The plac.go file helps you automate any site-specific network settings that you might require. The ability to update the Nessus plugins enables you to run the most recent vulnerability checks available. If you have to, you can write your own checks via NASL. In the end, you will find that you have an open source vulnerability scanner -recommended by Infosec practioners, SANS instructors, and used by many commercial companies – and this extremely portable toolbox can literally run from your pocket.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

**References:**

"Booting Up." The White Glove Tutorial." Fred Cohen & Associates Website. 22 Jan 2002
http://all.net

"The White Glove" Fred Cohen & Associates Website. 22 Jan 2002
http://all.net/WG/index.html

"Buffer Overflow in UPnP Service On Microsoft Windows." CERT® Advisory CA-2001-37. 20 Dec 2001.
http://www.cert.org/advisories/CA-2001-37.html

Deraison, Renaud. Nessus Project.
http://www.nessus.org/intro.html

van den Berg, Richard and Hugo van der Kooij. "Nessus FAQ" 11 Jan 2002
http://www.nessus.org/doc/faq.html#Q.OTHER.HOWDOES

Forristal, Jeff and Greg Shipley. "Vulnerability Assessment Scanners." Network Computing Magazine. 8 January 2001.
http://www.nwc.com/1201/1201f1b1.html

Fratto, Mike. "Security Awards." Network Computing. 14 May 2001.
http://www.nwc.com/1210/1210f111.html#vulnassess

Green, John. "Track 7 – Auditing Networks, Perimeters and Systems." Auditing Networks with Nmap and Other Tools. 31 January 2001.

"Guardent Announces Breakthrough 24X7 Managed Security Platform That Supports Open Source Security Technologies, Reduces Costs and Maximizes Performance." Guardent Press Release. 12 December 2001.
http://www.guardent.com/pr2001-12-12-01_SDA.htm

"New Scripts." Nessus Organization. 29 December 2001.
http://www.nessus.org/scripts.html

"Universal Plug and Play Script." Nessus Organization. 29 December 2001.
http://www.nessus.org/nasl/upnp_xp.nasl

"VIGILANTe Joins The Center for Internet Security." *The Encyclopedia of Computer Security*. IT Security Website. 27 February 2001.
http://www.itsecurity.com/tecsnews/feb2001/feb514.htm

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

**Resources**

General Nessus information can be found at: http://www.nessus.org

Nmap http://www.insecure.org/
Arping http://synscan.nss.nu/programs.php
Hping http://www.hping.org/
Icmpenum http://razor.bindview.com/tools/desc/icmpenum_readme.html
Xprobe http://www.sys-security.com/html/projects/X.html
Tcpdump www.tcpdump.org
Ethereal www.ethereal.com
Dsniff http://www.monkey.org/~dugsong/dsniff/
Ettercap http://ettercap.sourceforge.net/
Sniffit http://reptile.rug.ac.be/~coder/sniffit/sniffit.html

**Bootable Linux CD's:**
White Glove: http://all.net
Portable Linux Auditing CD: http://sourceforge.net/projects/plac/
Knoppix: http://www.knopper.net/knoppix/
Linux Care Bootable Tool Box: http://lbt.linuxcare.com/index.epl
LNX-BBC: http://www.lnx-bbc.org/
DyneBolic: http://lab.dyne.org/DyneBolic
Trinux: http://trinux.sourceforge.net/
Virtual Linux: http://sourceforge.net/projects/virtual-linux

© SANS Institute 2002, Author retains full rights.

Last Updated: June 19th, 2013

# Upcoming SANS Training

**Click Here for a full list of all Upcoming SANS Events by Location**

| | | | |
|---|---|---|---|
| **Digital Forensics & Incident Response Summit 2013** | **Austin, TXUS** | **Jul 09, 2013 - Jul 16, 2013** | **Live Event** |
| **SANS London Summer 2013** | **London, GB** | **Jul 09, 2013 - Jul 16, 2013** | **Live Event** |
| **SANS Rocky Mountain 2013** | **Denver, COUS** | **Jul 14, 2013 - Jul 20, 2013** | **Live Event** |
| **SEC528 SANS Training Program for the CompTIA® New Advanced Security Practitioner (CASP) Certification** | **Washington, DCUS** | **Jul 22, 2013 - Jul 26, 2013** | **Live Event** |
| **SANS Mumbai 2013** | **Mumbai, IN** | **Jul 22, 2013 - Jul 27, 2013** | **Live Event** |
| **SANS San Francisco 2013** | **San Francisco, CAUS** | **Jul 29, 2013 - Aug 03, 2013** | **Live Event** |
| **SANS Boston 2013** | **Boston, MAUS** | **Aug 05, 2013 - Aug 10, 2013** | **Live Event** |
| **SANS SEC 560: Network Penetration Testing @ Bangalore 2013** | **Bangalore, IN** | **Aug 05, 2013 - Aug 10, 2013** | **Live Event** |
| **Critical Security Controls Summit** | **Washington, DCUS** | **Aug 12, 2013 - Aug 18, 2013** | **Live Event** |
| **Industrial Control Systems Security Training - DC** | **Washington, DCUS** | **Aug 12, 2013 - Aug 16, 2013** | **Live Event** |
| **SANS Virginia Beach 2013** | **Virginia Beach, VAUS** | **Aug 19, 2013 - Aug 30, 2013** | **Live Event** |
| **SANS Thailand 2013** | **Bangkok, TH** | **Aug 19, 2013 - Aug 31, 2013** | **Live Event** |
| **SANS Melbourne 2013** | **Melbourne, AU** | **Sep 02, 2013 - Sep 07, 2013** | **Live Event** |
| **SANS Capital City 2013** | **Washington, DCUS** | **Sep 03, 2013 - Sep 08, 2013** | **Live Event** |
| **Network Security 2013** | **Las Vegas, NVUS** | **Sep 14, 2013 - Sep 23, 2013** | **Live Event** |
| **SANS Canberra 2013** | **OnlineAU** | **Jul 01, 2013 - Jul 13, 2013** | **Live Event** |
| **SANS OnDemand** | **Books & MP3s OnlyUS** | **Anytime** | **Self Paced** |