

Security Auditing Utility - NESSUS

Objective:

The objective of this lab is to learn to install, configure, and use an open-source security auditing tool. The utility of choice in this lab is NESSUS. It is the most widely used security auditing tool in the open source community. This lab will cover not only the installation and use of the utility, but also how to interpret the results. The benefits of using such a tool, as well as the limitations, will also be discussed.

References:

<http://nessus.org>
<http://install.nessus.org>
<http://nessuswx.nessus.org>

Equipment/Programs Used:

Dell Desktop
Nessus Security Scanner
NessusWX Windows Client
Linux RedHat 9.0

Procedures:

This section will describe the processes involved in getting a complete Network Security Scanning Server running using RedHat 9 and Nessus. It is written in such a way that a person should be able to follow the directions exactly as they are presented, and the system should work as expected. It is broken into the following sub-sections:

- RedHat 9 Pre-Install Requirements
- Install the Nessus Server
- Install the Nessus Client
- Scan the Local Network
- Analyze the Results

RedHat 9 Pre-Install Requirements

Although installing the Nessus server is fairly straight-forward, there are a few basic requirements that must be met. The install script used to install the server and client require that the shell scripting language be installed correctly. Also, if the user desires to run the Nessus client locally on the machine, then the GTK packages must also

be installed. Details for this can be found on the Nessus website. Along with the GTK, the machine must also have the Xserver running and some type of interface that is compatible with GTK-built applications (KDE and Gnome are both acceptable).

It may also be necessary to alter the firewall to allow Nessus unlimited ability to scan. The default port for Nessus is 1241. Thus the firewall can be altered to allow anything coming into or out of that port to pass. If the user is not familiar with configuring firewall rules, then it may be simpler to shut off the firewall temporarily by using the command “/sbin/service iptables stop,” and then restart the firewall after the scan is complete by issuing the command “/sbin/service iptables start.” This method is not recommended though, due to the security concerns inherent with turning off a firewall.

Install the Nessus Server

The installation of Nessus should be done under the privileges of a non-root account. Under an unprivileged account, execute the following command to install Nessus:

```
lynx -source http://install.nessus.org | sh
```

When prompted, enter the root password. Select the default for all of the questions asked. Be sure to pay attention when it asks about the proxy. If proxy settings are required, be sure to answer ‘yes.’ The script will then automatically download and install the necessary packages for Nessus, including both the server daemon and the client. This process may take a considerable amount of time, depending on the speed of the computer and internet connection.

Once this process has finished, the server must still be configured. First it is necessary to create a site certificate for the encryption:

```
/usr/local/sbin/nessus-mkcert
```

It is also necessary to provide a way for users to connect to the server. One of the benefits of Nessus is that it allows for multiple users, each with different passwords, and more importantly, with different privileges. For example, if one user should be restricted to only running an audit on a certain IP address range, that can be specified. Now add a user for nessus (this will be the username and password that the client uses to connect to the server):

```
/usr/local/sbin/nessus-adduser
```

Before running the daemon, it is a good idea to update the plugins. The plugins are the actual security audits that will be performed. Plugins are submitted by security professionals and individuals from around the world. This helps to maintain an open-source, but highly dynamic and extensible security auditing tool. Update the plugins frequently by issuing the following command:

```
/usr/local/sbin/nessus-update-plugins
```

Now start the daemon:

```
/usr/local/sbin/nessusd -D
```

Install the Nessus Client

The Nessus client is already installed on the same machine that the server is running on. The client can be invoked by simply issuing the command “/usr/local/sbin/nessus.” To install the client on a Windows machine, download the NessusWX client from the Nessus web page. After unzipping the package, simply open the Nessuswx.exe file and this will start up the client.

Scan the Local Network

Now we can test the install by running the client. This is done by simply typing “nessus” at the shell prompt (you may need to type /usr/local/sbin/nessus). Once the GUI for the client displays, then use the first screen to establish a connection to the server. The host should remain ‘localhost’ the port should remain the default (1241), and enter the username and password that was specified during the ‘adduser’ portion of the install. Click the ‘login’ button and accept the certificate.

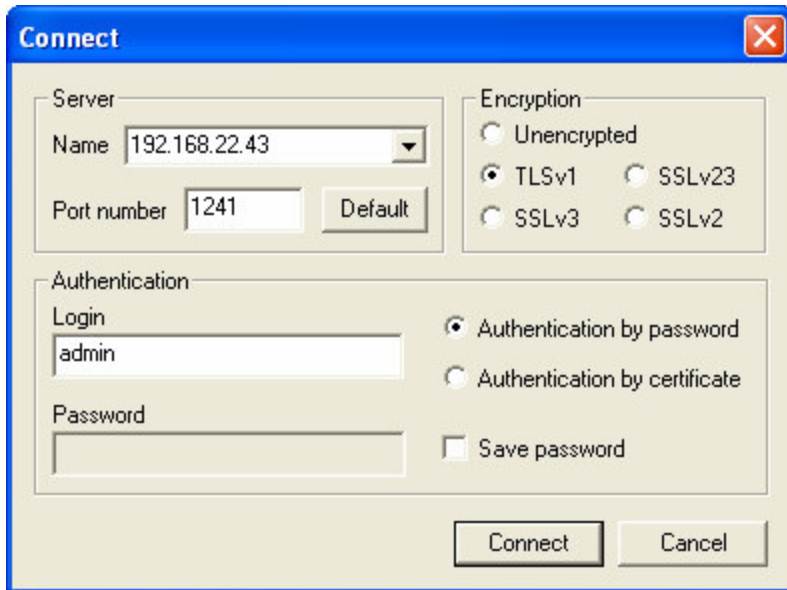
The next step is to decide what plugins to run. For the first scan, it may be best to just use the default, which is to enable all plugins except the ‘dangerous’ ones. Then click on the “Prefs” tab to mark any specific preferences that may be desired or needed to run in the scan.

Add the hosts that you would like to scan (be sure that these are computers you have a legitimate right to scan). Then run the scan. It may take some time for this process to complete, depending on the number of machines discovered, and the services running on those machines.

As an example, a step by step illustration of how to run an audit using a Windows client is provided. For this example, the IT Network at BYU will be scanned for vulnerabilities. The first step is to run the client by opening the “Nessuswx.exe” file that came in the zipped package.



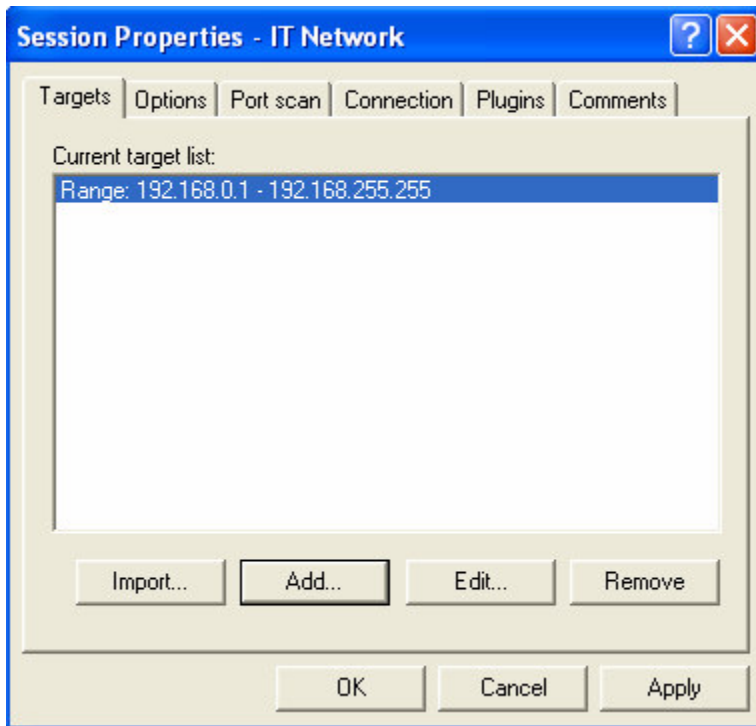
It may make comments about a database check that needs to be run. Simply click ‘okay.’ The database capabilities will not be used for this example. Then click on “communications>Connect” and this will bring up a dialogue asking for information on how to connect to the server. Enter in the IP address of the server, and also the username that was chosen.



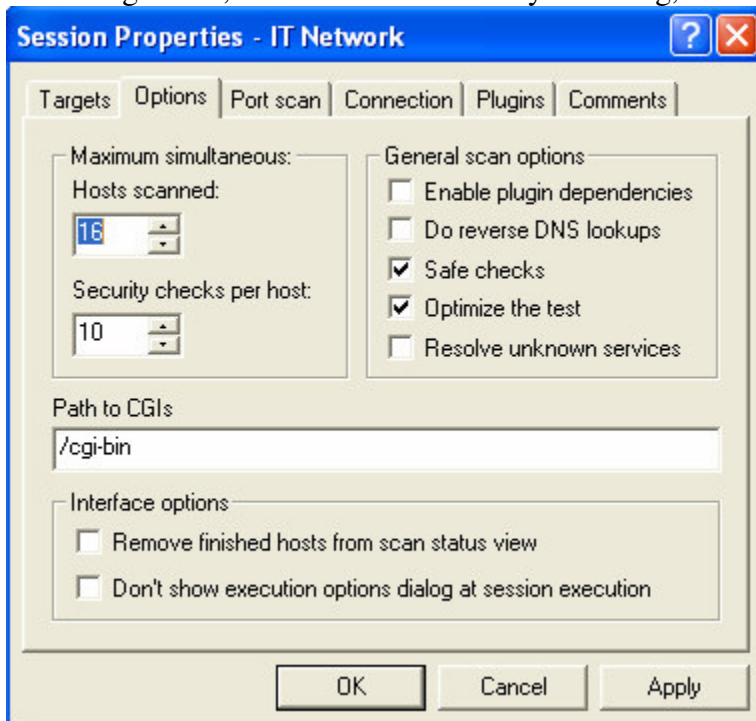
Be sure that a supported encryption type is chosen for the scan. If the server machine does not support an encrypted link through SSL or TLS, then select the “Unencrypted” option. Then click on “Connect”. It should then prompt for the password for the username that was entered. If this is correct, then the session will be established, and it will begin to download information about the plugins that are currently available. Now a new “Session” will be created by clicking on “Session>New”. This will bring up a dialog that will ask for the name of the Session. Enter in any name that is descriptive of the session. Be sure the “define additional properties” checkbox is checked, and then click on “Create”.



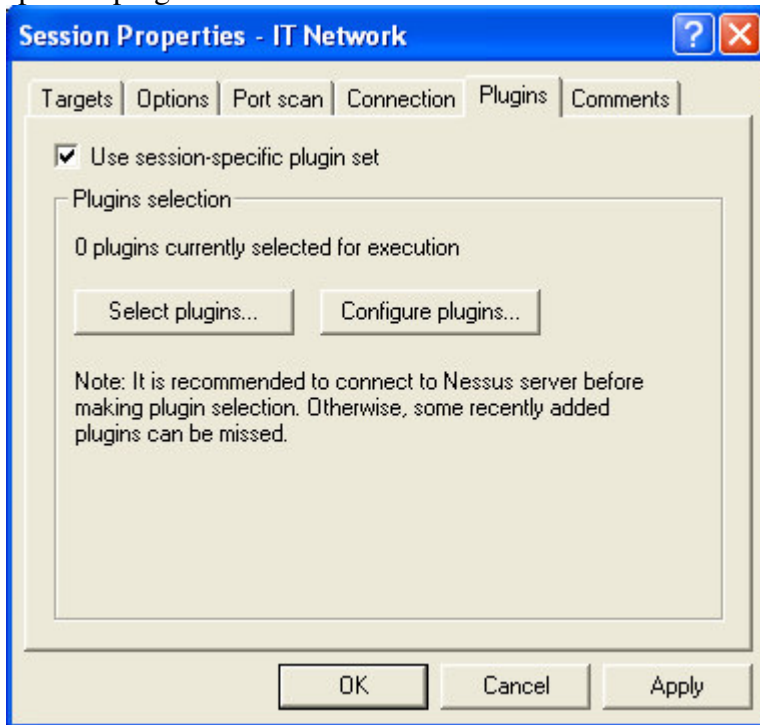
This will bring up a dialogue that allows for multiple configurations. On the first screen select the range of IP addresses that will be included in the scan. This is done by clicking on “Add” and following the dialogue instructions.



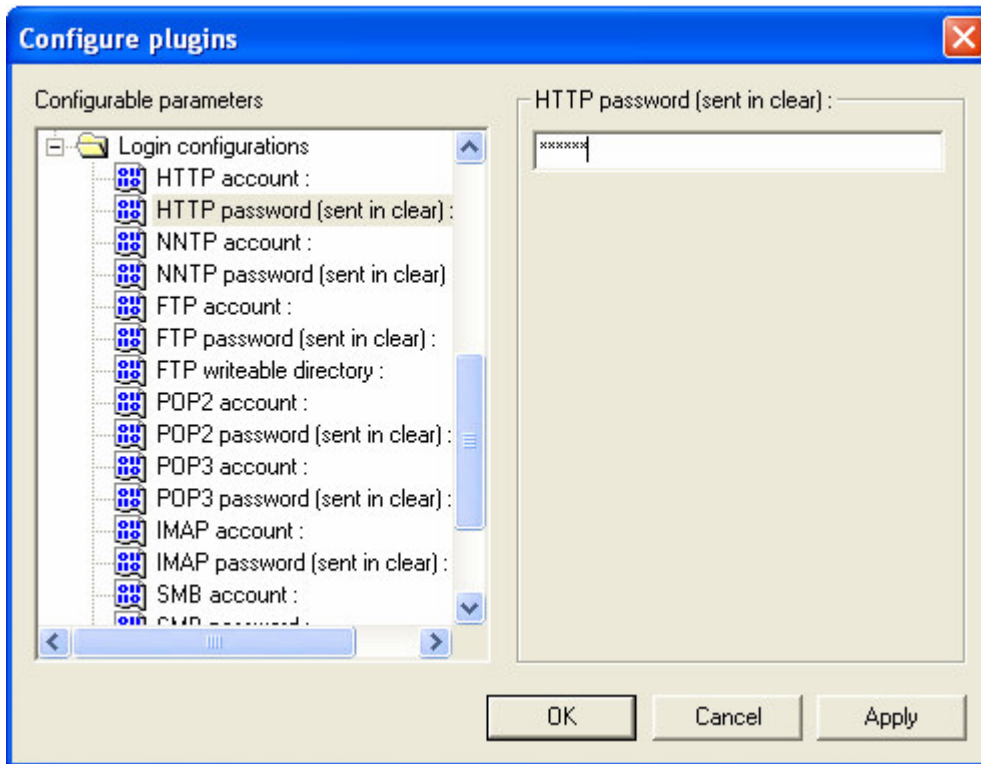
Next, configure the options for the scan. Click on the “Options” tab in order to view the available options. The defaults are adequate for this example. However, it is important to note that the check box labeled “Safe checks” is very important. By checking this box, any plugin that may cause the server to crash or be altered in any way will be disabled. This may raise some extra false positives, but unless the user is skilled in running audits, and is sure of what they are doing, this box should be left checked.



The “Port Scan” tab will allow the user to specify the range of ports that will be included when looking for services and open ports. The default will be used in this example. The next thing that we need to configure is the plugin preferences. This is done by clicking on the “Plugins” tab and checking the box marked “Use Session-specific plugin set.”



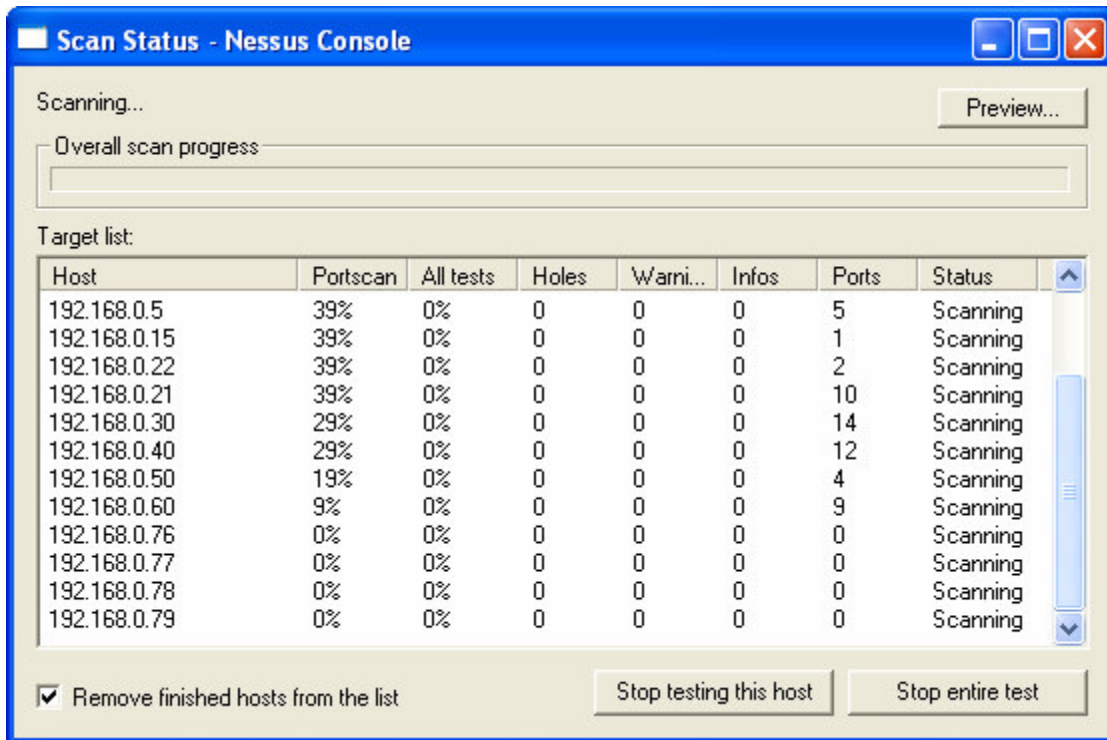
Then click on the “Select plugins” to select the plugins that are needed for the scan. For this example, all safe plugins were selected. After selecting the plugins, click on the “Configure plugins...” button. This will allow the user to select preferences for each plugin that may allow for modification. Go through the list and modify any preferences that are desired or necessary for the scan to be accurate.



Once the plugins have been configured, click on “OK” for all of the dialogue boxes that are still open. This should now produce an icon similar to the following:



To run the scan, simply double-click on the icon, and select “Execute.” This will bring up a display that will show the real-time progress of the scan. An example of this is shown below.



The scan could take a considerably long time, depending on how many hosts are being scanned, and how many services are running on the hosts. Once the scans are complete, then another dialogue is displayed that shows all of the scans that have been completed for this session. This is very useful, especially when action is taken to improve security. Additional audits can easily be compared to previous audits to elaborate on what security threats still exist on the system.

Analyze the Results

The final step in running the report is to view the report and mark false positives. By double-clicking on the completed scan, it will bring up a list of the warning and alerts that were produced. Any known false positives should be marked as such, so that they can easily be left out of the final report. The final step then is to adjust the report to fit any standards that may be required, and then to print the report. This report should then be used to upgrade the network and to provide a guideline for fixing some of the problems discovered on the network. Further analysis of the results will be provided in the “Results” section of the write-up.

Results:

The Nessus Security Scanner was successfully installed by following the instructions as outlined in the procedures section of the document. There were a total of 978 alerts generated by the network. Of those, 62 were considered high severity, 618 were low severity, and 298 were informational. The scan took 48 minutes and 48

seconds to complete. The most vulnerable host seems to be 192.168.0.30, with 7 high severity alerts and 54 warnings. The report of the results was generated in pdf format, and is included as an additional document along with this report. The included document only contains a detailed report of the 22 highest rated alerts.

Conclusions:

A Network Security Scanner is a vital element of any network where constant auditing is considered a necessity. Using a product such as Nessus helps to eliminate some of the financial burden associated with alternative commercial products. The Nessus scanner has multiple features that make it a very powerful utility. One of these is the fact that it can alternate between both active and passive probing methods. To elaborate on the difference between these two types, a simple explanation will be provided. In an inactive or passive scan, a plugin will probe the server or device for a known vulnerability based on header information or banner information. However, in an active scan, the same plugin will actually attempt to exploit the vulnerability in order to eliminate a false positive that might show up in an inactive scan.

Nessus also provides some excellent features that a more commercialized product does not. One of these is the concept of a server/client program. Most (not all) commercial security scanners are based on just a client that scans a network. Using the server system that Nessus provides allows for multiple user accounts, each with distinct permissions and rule sets. Additionally, the Nessus plugin database is something that is updated daily and is contributed to by thousands of security experts around the globe. Commercial products must keep their own vulnerability database, and will often charge money for updates. Nessus provides this service for free, and has a large group of individuals who help to contribute to the progress of the utility.

In conclusion, the Nessus tool that was used in this lab is an excellent way to audit a network for known vulnerabilities. It is versatile, powerful, and inexpensive to deploy. However, it is not a 'cure-all' for security. Although this is a powerful tool that can aid in the discovery of vulnerabilities, there are still other measures that must be taken in order to improve security. Some of these other things include employee and user training, and software or application firewalls. Despite some of these limitations of the security auditing tool, it can still be easily said that it "fills the measure of its creation" by providing administrators and security personnel with a marvelous auditing tool.