MSIS Project: VULNERABILITY ASSESSMENT

Metalsoft Vulnerability Assessment

Jason Ekstrom

Lewis University

Abstract

I work for a company called Metalsoft Inc. Metalsoft Inc. makes software for programming turrets, lasers, and press-brakes in the manufacturing industry. These machines are for cutting and bending sheet metal. Metalsoft has two sites located in North America. The headquarters is located in Orange County, CA. In Itasca, IL Metalsoft has a sales office. At Metalsoft's headquarters there are around 80 employees while the Itasca office consists of three employees. Once a month Metalsoft holds a training school at the Itasca site where a trainer from California comes in and uses the facilities. Each of the salesmen has their own laptop and occasionally travels. They all connect to a single server that holds their ACT! database for sales leads and stores their quotes. This server is used as a file server, print server, DHCP, DNS, FTP server, and occasionally a web server. The only time we use our site for a Web Page is when the California Web Site goes down and we allow our customers to be redirected to our site to receive patches or software updates. Most of the salesmen uses for the server are for when they are in the office. Our VP of sales needs to login to the server using Terminal Services to check the ACT! database when he is at home or traveling. I need to login to the server using Terminal Services to do system administration. The Itasca office is connected to the headquarters in California through a VPN to check databases there that include customer trouble tickets and more sale leads. The server is running Windows 2000 Advanced Server that is running Terminal Services and IIS 5.0. This server is running FileZilla Server 0.9.19 for the FTP. All of the salesmen have Windows XP SP2.

Until recently, the Metalsoft Sales office was protected by a firewall. The Itasca office did not have anyway of knowing what was happening on the network or where the vulnerabilities were. Everything was working properly but we did not know if something was happening that we could not see. From viewing Metalsoft's FTP logs there were people who should not have access to the FTP server and were trying to gain access. Even though they were unsuccessful and ban rules had been put in place for them, Metalsoft did not know what else these malicious people were trying to do to the server.

The purpose of this project is to perform a vulnerability analysis for Metalsoft Inc. in Itasca. In order to accomplish this I did a vulnerability scan using Nessus 3.0.6. Nessus is free software that has predefined attack modules and you can write your own modules to perform attacks and see if a computer is vulnerable. Once vulnerabilities were discovered, I tried to reduce the number of vulnerabilities as much as possible. Then Metalsoft needed to know what normal traffic was like on the network and what kind of traffic was taking place. To do this I used Wireshark 0.99.6a. Wireshark is a free software using the GNU GPL licensing and is used for traffic monitoring.

Table of Contents

Abstract…… 2
Introduction……..... 5
Nessus Vulnerability Scanner…… 8
Prerequisites…….. 8
Deployment…….. 8
Getting and Registering for Nessus…….. 8
Installation……..... 10
Registering……..... 11
Direct Feed……..... 12
Nessus Interface……. 12
Start Scan Task……. 13
View Reports……. 17
Manage Policies……. 20
Update Plugins…… 26
The Experience at Metalsoft…….. 26
Metalsoft Network Vulnerability Scan…….. 26
Server Scan……. 27
Router……. 28
Salesmen Computers and Printers…….. 29
Other Experiences with Nessus…….. 30
Nessus Review…… 31
Wireshark…… 32
Prerequisites…… 32
Deployment……. 33
Installation……... 33
Wireshark Procedure……... 38
-a duration:43200……. 40
-b filesize:102400…… 40
-I 2……... 40
-N nmtC……... 41
-t ad…….. 41
-w d:\users\jasone\Metalsoft_Night_Batch……. 41
Reviewing Results…… 41
Review of Wireshark…… 45
Conclusion……... 46
References…… 47

Table of Figures

Figure 1: Metalsoft network configuration at Itasca……..... 6
Figure 2: Nessus registration page…….. 9
Figure 3: Selecting a platform-specific version…….. 9
Figure 4: Nessus activation code…… 10

Figure 5: Nessus running as Windows service…….. 11
Figure 6: Registering for a license after installation……... 11
Figure 7: Registering the software……. 12
Figure 8: Nessus welcome screen on Windows……. 13
Figure 9: Nessus start scan task…….. 14
Figure 10: Configure plugins…… 14
Figure 11: Identify the host server…… 15
Figure 12: Configure the Nessus port……..... 16
Figure 13: Nessus scan progress……... 16
Figure 14: Nessus sample report……..... 17
Figure 15: Nessus plugin information…… 18
Figure 16: Plugin source code……..... 18
Figure 17: Nessus view reports screen……. 19
Figure 18: Compare two reports example…… 20
Figure 19: Manage policies screen……... 21
Figure 20: Nessus manage policy settings…… 21
Figure 21: Ping tab of change settings……. 22
Figure 22: Change settings services tab…… 23
Figure 23: Change settings credentials tab…….. 23
Figure 24: Change settings web tab……. 24
Figure 25: Change settings compliance tab……. 24
Figure 26: Change settings others tab…….. 25
Figure 27: Change settings plugins and options…….. 25
Figure 28: Update Nessus plugins screen…… 26
Figure 29: Hole in iisadmin…… 28
Figure 30: Printer vulnerabilities……. 29
Figure 31: Customer error messages…… 30
Figure 32: Software open ports…… 32
Figure 33: Wireshark welcome screen…… 33
Figure 34: GNU GPL license agreement…… 34
Figure 35: Wireshark tools and extensions……. 35
Figure 36: Wireshark shortcuts and file extensions……..... 36
Figure 37: Wireshark install directory……. 36
Figure 38: WinPcap installation page…….. 37
Figure 39: Wireshark finish installation…….. 38
Figure 40: Wireshark task manager memory usage……..... 39
Figure 41: Tshark running…..... 40
Figure 42: Wireshark main screen……... 42
Figure 43: Expression window…… 42
Figure 44: FTP being attacked……. 44
Figure 45: China attack…… 45

Introduction

We are living in an age where computers are running our lives both professionally and personally. This has made life more convenient for some in that they can now better communicate, transfer data faster, and make transactions at all hours of the day. With this convenience greater risks are apparent. Businesses are greatly impacted by this as they can have computer services that run twenty-four hours which means they have to keep ports open on their network. There are malicious people who are looking to attack businesses in order to acquire information or to take them off-line and disrupt business. Since traffic is allowed on the network at all times of the day, businesses have a need to know who and what is traveling across their network. There may be policies that state when and how the data can be accessed, but how do we know that everyone is abiding by these policies. System Administrators need to create a system to prevent unauthorized access, discover vulnerabilities, and keep track of what is happening on the network. This is where companies will need to implement a vulnerability assessment. There are many vulnerability scanners that are available today such as Nessus and SARA. These two examples are free and can be run from a host computer across the network. These programs have attack modules built into them and test each computer for vulnerabilities and report back to the host computer what attacks were successfully used. Once the vulnerabilities are known on the network we can address these problems so that malicious users do not exploit them. Once the vulnerabilities are resolved, we can use a network packet analyzer to monitor the traffic that is taking place over the network. There are free network packet analyzers available over the Internet. Some of these include Wireshark and Tcpdump. With network packet analyzers in place we can see packets on the network and see what ports are being used and where they are coming from. By using these types of software we have better knowledge of what is happening on the network and minimize the chances of attack. The logs can also be used for forensics in case a cyber crime takes place.

Previously Metalsoft Inc. did have any way of knowing how secure their sales office was. The sales office is located in Itasca, IL. This site contains their VP of Sales and two executive salesmen. All of them use laptops on the network and there are two network printers. All the salesmen travel and are in and out of the office. Occasionally the sales office will have a visitor from California who comes on our network when he has to teach a software class at the Itasca site. The other computers for the school are not attached to the network. While in the office the salesmen connect to a single server. This server holds their ACT! 6.0 database where sales leads are kept. The server also contains their sales quotes and any other data they need to backup. The server is used for file sharing, printer sharing, DNS, DHCP, and an FTP server. The FTP server is primarily used by myself for software backups and large data files I need to send to other employees. I create accounts for the employees temporarily to get their files and then delete their accounts as they do not need access to this server. When our main website in California goes down our server is used as a web page for customers to download patches and updates for the software Metalsoft supports. All of the salesmen computers', server, and printers are connected to a single switch. The switch sits behind a VPN router that is connected to our headquarters in California. This router sits behind a Cisco firewall. See Figure 1.

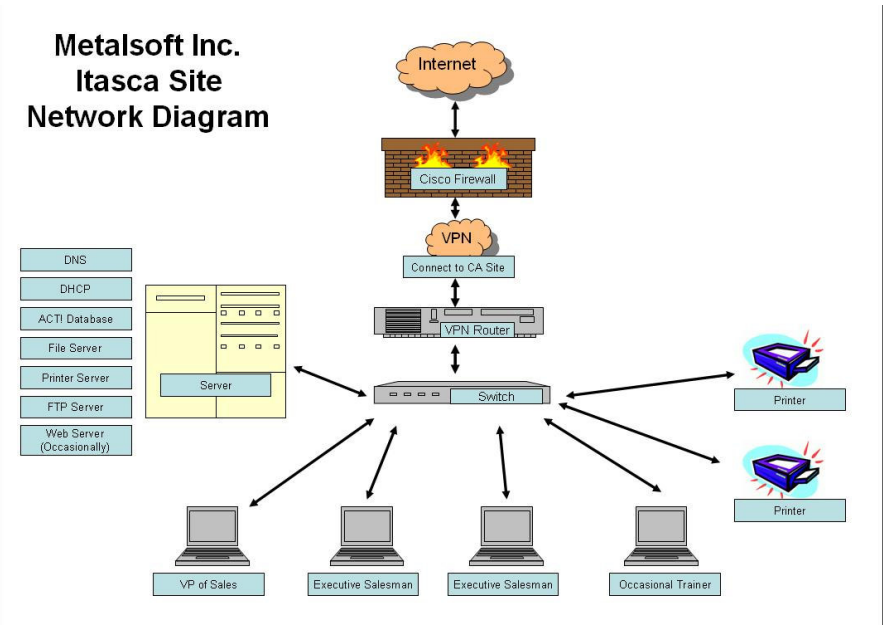


Figure 1: Metalsoft Network Configuration at Itasca

I used to work at the Itasca site but was moved to Mount Prospect, IL. I have been assigned as a System Administrator and take care of the server remotely using Terminal Services. The Itasca site did not have any vulnerability assessment. My purpose for this project was to monitor traffic on the network and discover any vulnerability. From logs on the FTP server, there were people who were trying to connect to the FTP using dictionary attacks. One of our salesmen's computers collects a lot of virus and spy-ware and we needed to make sure that no bad traffic is coming in or going out.

We deployed two types of software to accomplish this. All of the software has been installed to a Windows 2000 Advanced Server as I have remote access to this computer and can run them from there. Since the network is so small and we do not have a lot of traffic, the server is capable of handling these types of scans.

First we needed to check the stability of the computers, printers, and router on the network. To scan for vulnerabilities we ran Tenable Nessus. At first, we used version 3.0.5, and over time version 3.0.6 was released. We ran Nessus to minimize as many vulnerabilities as possible before we started to watch traffic on the network.

Secondly, we used Wireshark to monitor network traffic. Wireshark is the replacement for Ethereal. We started using version 0.99.5 and eventually version 0.99.6a was released. We used Wireshark for a few months to get an idea what normal traffic is like on the network and collect information that was traveling on the network. We needed to know who was trying to get into the network from the outside.

The following paper will describe each of the software in detail. I will explain my experience with the software from installation to the current state that it is running at right now.

Nessus Vulnerability Scanner

According to Tenable, the company that supports Nessus, "The Nessus vulnerability scanner, is the world-leader in active scanners, featuring high speed discovery, configuration auditing, asset profiling, sensitive data discovery and vulnerability analysis of your security posture." (Tenable Network Security, The Network Vulnerability Scanner, para. 1) Nessus is a free vulnerability scanner that can be used by Windows, Linux, and Unix Operating Systems. Tenable offers the largest vulnerability knowledge base to support even the most complex environment. Some of Nessus' benefits include:

- Intelligent Scaning – Nessus does not assume that all your services are running on their fixed port so if you are using services on different ports Nessus will detect and safely test the security appropriately.
- Modular Architecture – This provides a client/server architecture that allows system administrators the flexibility to deploy the server that will perform the scans and give multiple people the ability to use the client software to run the tests from their own PC. This offers more flexibility for administration and configurations.
- CVE Compatible – All of the plugins link to Common Vulnerabilities and Exposures (CVE), which is a web site that holds published security vulnerabilities, for system administrators to retrieve more information of the vulnerability. These vulnerabilities will also include references to CERT, Bugtraq, and other vendor security alerts web sites.
- Plug-in Architecture – All of the security tests are written to their own plugin. Because of this, if system administrators want to write their own tests they do not have to write to the Nessus server engine.
- NASL – NASL (Nessus Attack Scripting Language) is the scripting language the plugins are written in to perform security tasks. The security checks can also be written in the C programming language.
- Up-to-date Security Vulnerability Database – To keep up with current security threats, the Tenable Knowledge Base is updated daily and Nessus offers a plugin update feature. Downloads can also be found on FTP servers and mirrors.
- Tests Multiple Hosts Simultaneously – This allows the Nessus server to perform the tests on multiple hosts at one time.
- Multiple Services – If you have two of the same service running on a server, (for example, two web servers running off of port 80 and port 8080), Nessus will test both services.
- Tests Cooperation – The security tests that are performed by Nessus do not perform unnecessary checks. For example, if the FTP server does not accept anonymous logins, then the anonymous login security checks will not be performed.
- Complete Reports – Nessus will report back to the system administrators the vulnerabilities, their risk levels, and information how to prevent them from being exploited.
- Full SSL Support – Nessus can perform security tests over SSL, which stands for Secure Sockets Layer that provides cryptographic protocols for secure communications. You can even supply Nessus with certificates so that the tests can integrate with a PKI environment. PKI stands for Public Key Infrastructure that binds public keys and identifies authority using certificates.

- Smart Plugins – Nessus can determine which plugins should and should not be used against a remote host but this option can be turned off.
- Non-Destructive – If a security check is going to cause harm to a server that can not have any down time you can perform "safe checks" that will rely on banners, which are messages that the attack could take place, rather than exploiting the vulnerability.

Prerequisites

The hardware requirements depend on the size of the network. If you have a large network that is not a Class C then you may need to have a machine with 1-4 GB of RAM. For Windows environments Nessus recommends that you install on a server family operating system to prevent false negatives. For example, if Nessus is installed on a Windows XP SP2, there may be some problems registering certain DLLs. For a Class C network you need a machine with at least:

- Pentium III processor with 733 Mhz
- 256 MB RAM
- At least 1 GB of free Hard Drive space

Metalsoft's server hardware is:

- Pentium IV processor with 2.8 Ghz
- 1 GB of RAM
- 30 GB free Hard Drive Space

Deployment

Where you deploy your Nessus Server can have a great impact on the scans. You must have knowledge of where routing, filters, and firewalls are located. You must have good IP connectivity. Deploying behind a NAT device may give undesirable results unless you just plan on scanning the internal network. NAT stands for Network Address Translation and is a technique for network traffic through a router to change the IP address from a public address to the network's private addresses. Having to pass through a NAT, firewall, or proxy device can distort the scan as a firewall could prevent or hide probes from a Nessus scan.

For Metalsoft's deployment we put Nessus on our Windows 2000 Advanced Server since we were just interested on scanning computers that are within the network. The hardware requirements of the server exceed the minimal requirements and we the network only has three consistent users. With Windows 2000 Advanced Server we do not need to worry about the Windows Firewall affecting the scans and since this is in the Windows server family, all of the DLLs should register properly.

Getting and Registering for Nessus

To obtain a copy of Nessus you can go to http://www.nessus.org/download/. From there you will be asked to accept a license agreement and then you have to fill out a form giving your name, title, and email address. You can provide the same information more than once and get more downloads for different Operating Systems or if you have lost your installer. The registration page is shown in Figure 2, while the operating system selection is shown in Figure3.

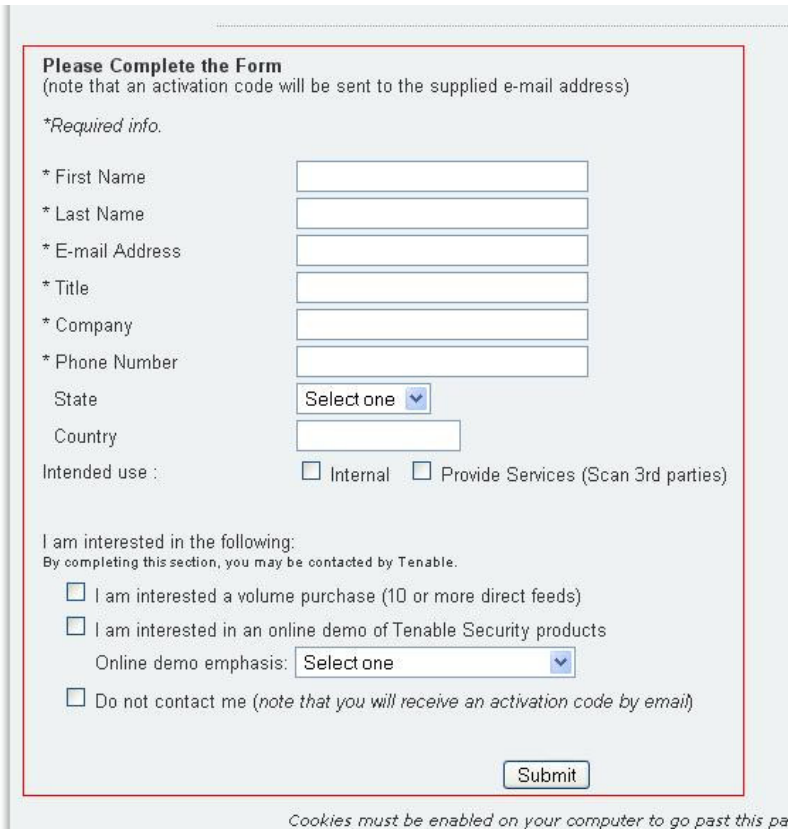


Figure 2: Nessus registration page

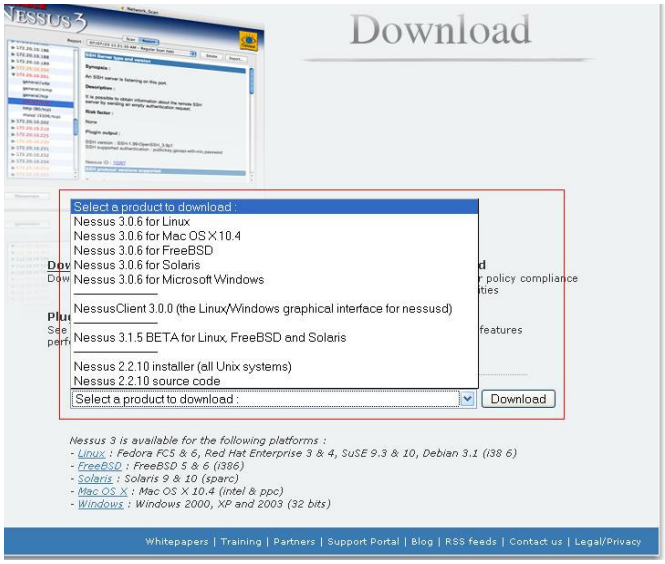
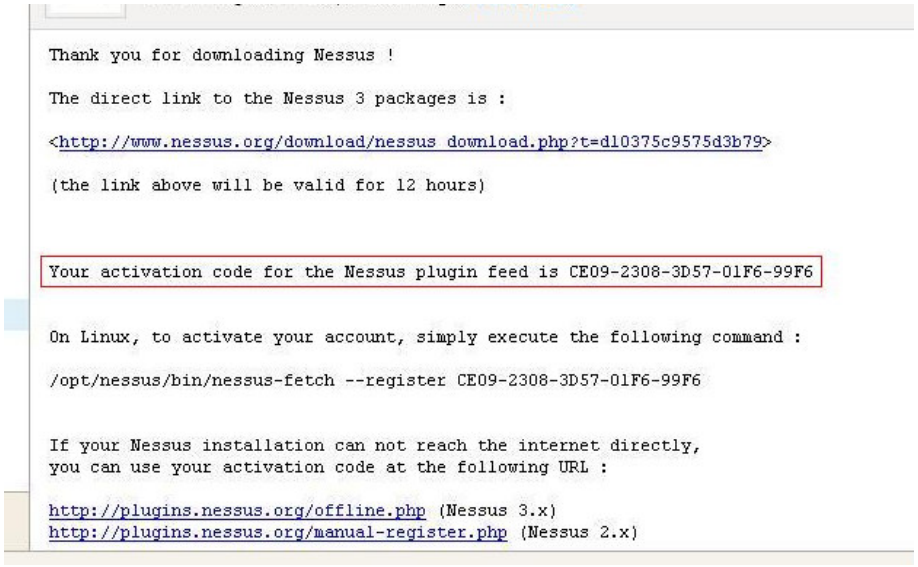


Figure 3: Selecting a platform-specific version

While doing this process you will be emailed immediately a registration code to activate your Nessus installation. You will have a link that is open for 12 hours to make anymore downloads

needed. The registration code can be done at any time. The code only works once so if you install Nessus to another machine you will need another code. Figure 4 has a sample email from Tenable:



Thank you for downloading Nessus !

The direct link to the Nessus 3 packages is :

<http://www.nessus.org/download/nessus_download.php?t=d10375c9575d3b79>

(the link above will be valid for 12 hours)

Your activation code for the Nessus plugin feed is CE09-2308-3D57-01F6-99F6

On Linux, to activate your account, simply execute the following command :

/opt/nessus/bin/nessus-fetch --register CE09-2308-3D57-01F6-99F6

If your Nessus installation can not reach the internet directly,
you can use your activation code at the following URL :

http://plugins.nessus.org/offline.php (Nessus 3.x)
http://plugins.nessus.org/manual-register.php (Nessus 2.x)

Figure 4: Nessus Activation Code

Installation

Originally we installed Nessus 3.0.5 but then upgraded to Nessus 3.0.6. We did not have release notes to see what was fixed from the previous version, but we have not found any new features in the software. Recently Tenable has released Nessus 3.0.6.1 but we have not used this version. Over the last six months Tenable has released three updates of Nessus. To update a Nessus installation is the same as installing the software originally. The setup executable will overwrite the software system files but all your settings and reports remain for a Windows environment. I have installed this to Windows XP SP2, Windows 2000 server and professional, Windows 2003, and Debian 4.0r0 and I have never had a problem with the installation. When doing the installation you should have an Internet connection if you plan to register. There are options to register if you do not have an Internet connection by downloading a PHP program from Tenable. To install Nessus is straight forward:

1. Run the Setup.exe
2. Agree to the License Agreement
3. Select where you want the files to install to
4. The files will begin to copy
5. You will be asked to register, or you can register later
6. The Plugin Update will run
7. Click Finish to end the installation

With the Windows version of Nessus, The server and GUI is installed on the server PC. If you are planning on using another station to run scans with the Nessus server you will need to install the client version of Nessus for Windows on that PC. The Nessus client for Windows installs the

same way as the full install, but you do not need to register the software as the software does not work without a server. With the client you can setup a certificate to run the server's service or you can add users who can access the server and give them privileges as to what scans they can run. If you install Nessus to Linux, at least I know for Debian, you will need to install Nessus and have the Nessus Daemon running. In order to use Nessus you will need to install the client as well on the server. These two packages do not come together for a Linux installation. In Windows you can find the Nessus Service running in Administrative Tools Services, as shown in Figure 5.

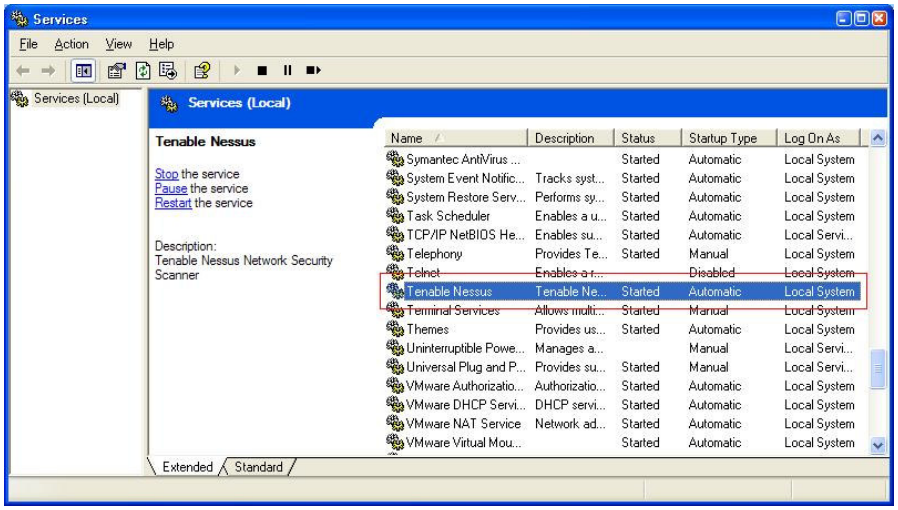


Figure 5: Nessus running as a Windows service

Registering

The free registration option allows us to download plugin updates. If you do not register, you can still use Nessus but you will not be able to get new plugins after the initial install. Since the registration is free we will have a 7 day delay on getting the latest updates so any new plugins created today will not be available for us to download until one week. If you installed Nessus and did not register, there is a link in the software to go to Tenable's site and register for license. You will be asked to provide your email address, as shown in Figure 6.

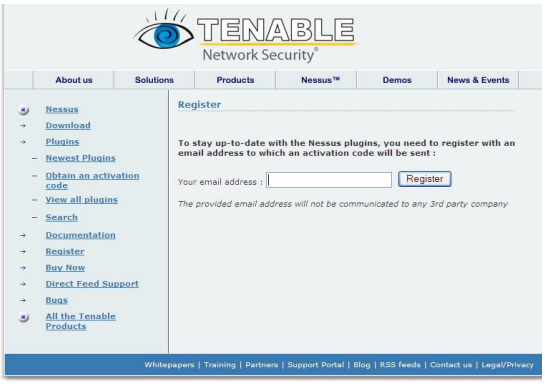


Figure 6: Registering for a license after installation

Once you have a code you can enter the code during the installation or there will be a link to register in the software, as shown in Figure 7.

Figure 7: Registering the software

Direct Feed

Direct Feed is an option you can purchase from Tenable. A seat for this option costs $1200. With a Direct Feed license you can get the following options:

- The newest plugins as soon as they are available
- Perform Compliance Checks that are agent-less configuration audits of UNIX and Windows systems. If your organization has configuration policies for servers or computers, Nessus can check to see if these systems comply. Nessus can also create an audit check for public and government best practices from NSA, CERT, and CIS.
- Create plugins to discover and test SCADA devices.
- Nessus can log into systems and audit the content on the machine for Adult Content, Credit Cards, Corporate Financial Spreadsheets, Drivers License Numbers, Document keywords, Human Resource information, Social Security numbers, and more.
- Support for operation and technical assistance. This includes the deployment and use of the audit policies, compliance checks, and content checks.

Nessus Interface

Figure 8 shows the welcome screen for Nessus for Windows. The welcome screen will give you all the options you need to begin using Nessus. This includes downloading plugins, running scan tasks, and management of the software such as the address book or custom policies. The welcome screen is easy to use and is straight forward with the options available. This screen will load if the Nessus service is not running but as soon as you perform a scan you will be prompted that the client server can not connect to the server.

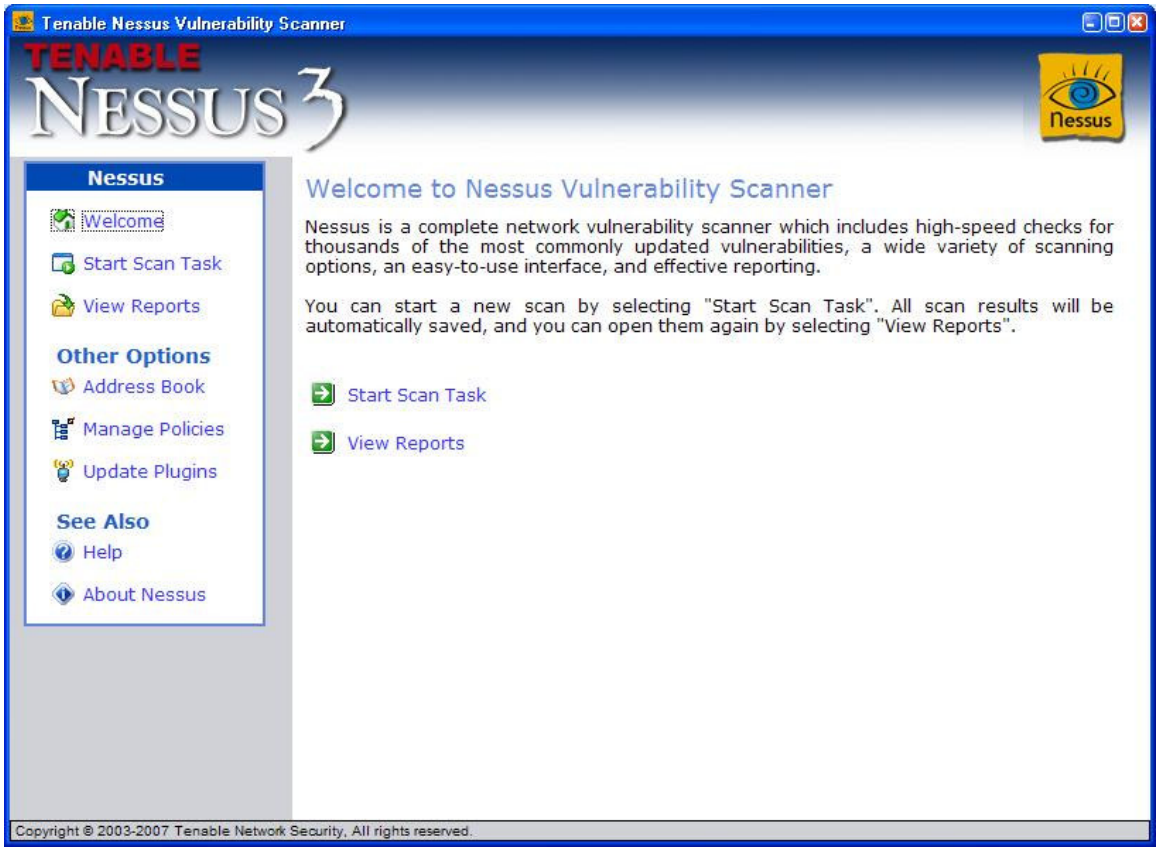


Figure 8: Nessus welcome screen on Windows

Start Scan Task

The first option is to start a scan task. This is the primary activity associated with Nessus. A wizard guides you through three screens of configuration options, helping you set the scan.

Figure 9 demonstrates how you identify the targets to scan. You can manually put in what computers you want to scan. You can use localhost to scan the Nessus server. You can input DNS names of the computer or the IP address. Each computer must be separated by a comma. If you want to scan a range of IP addresses you can put the beginning IP address, then a dash, and then the final IP address for a range. With the address book we can put in computers who are consistently looked at and by using the drop down button they will be listed to put in the scan task. The final option allows the user to import a text file of a list of computers to scan. You can give each computer their own line in the file and if there is a computer that you do not want to scan at that time, you can comment the computer out by using a '#'. An example would be:

#localhost
ComputerName
192.168.0.12

With this setup, ComputerName and IP address 192.168.0.12 will be scanned and localhost will be ignored.

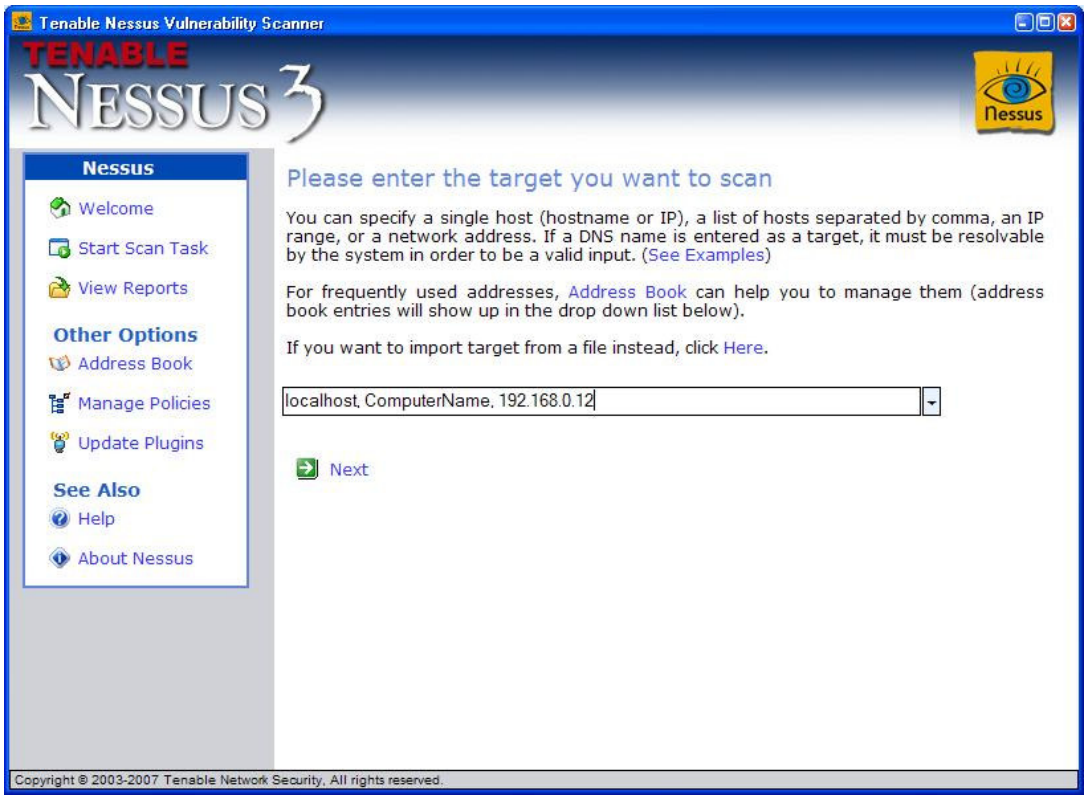


Figure 9: Nessus start scan task

After you select what computers you want to scan you come to the screen as to what plugins you want to use. Figure 10 demonstrates this step.

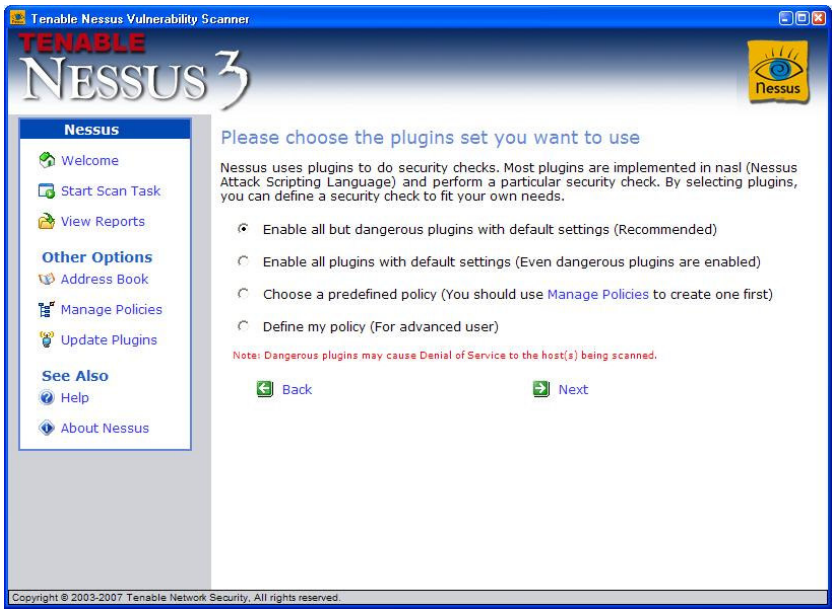


Figure 10: Configure plugins

The plugin screen provides a number of options.

- The first option is the safest. This goes through all the plugins but does not perform tests such as DoS that could harm your servers that are already warning. This performs tests that possibly were made for Linux distributions and can cost you some time.
- The second option will be a full attack on the server checking to see if the server can go down. This does not mean that the server will go down as Nessus does have safe checking but there is a possibility the server may go down. (I have experienced this, and I will describe this unfortunate situation later.)
- The third option will run a predefined policy that you have made before the scan. I will cover this in more detail later.
- The last option is similar to predefined policy as you will be asked what scans you want to run and what settings to use.

In Figure 11 we are asked which Nessus server we would like to run from.

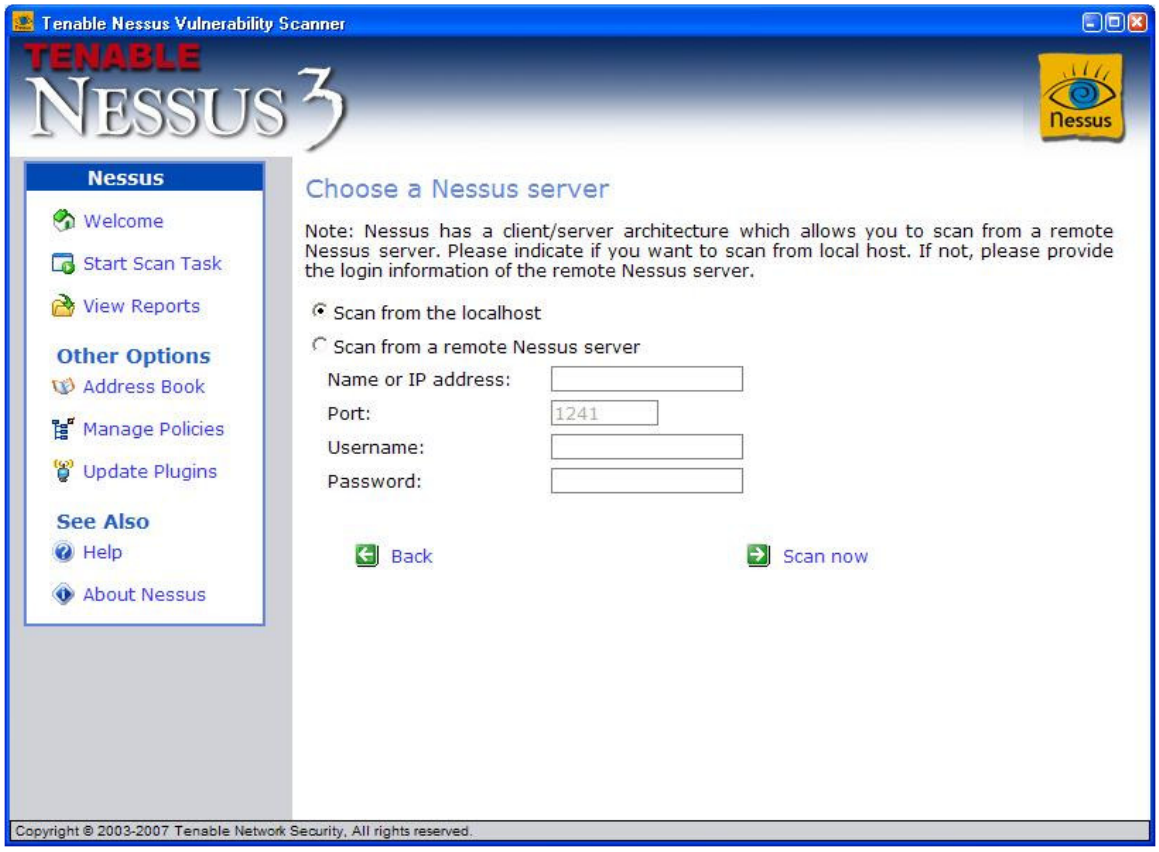


Figure 11: Identify the host server

If the computer is the Nessus server you can select localhost. If this is a client, then you need to put in the name of the computer or the IP address. Depending on how you authenticate to the server, you would either need to put in your username and password or you can ignore this if you use certificates. You will also need to make sure the correct port is listed. Port 1241 is used by

default. To change the port Nessus runs from you need to go to the PC that is running Nessus as a service and go to C:\Program Files\Tenable\Nessus. In here there is a file called ServerConf.exe that can change the listening port. Figure 12 shows how to set the server and Nessus port.

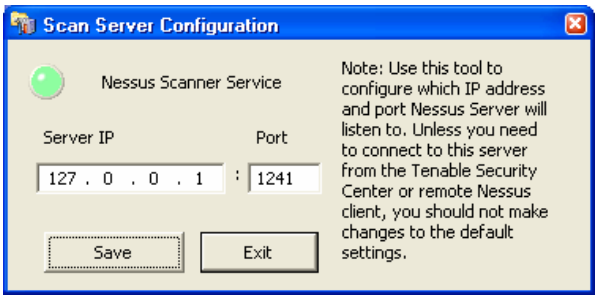


Figure 12: Configure the Nessus port

Once all the settings are defined, you can begin your scan. As Nessus is scanning you will see a screen that lists the computers being scanned. This gives you a progress bar that shows how far along Nessus is with the scan. The window will also show you how many ports are open, how many notes Nessus is recording, any warnings, and holes that Nessus finds during the scan. We can see an example of this in Figure 13.

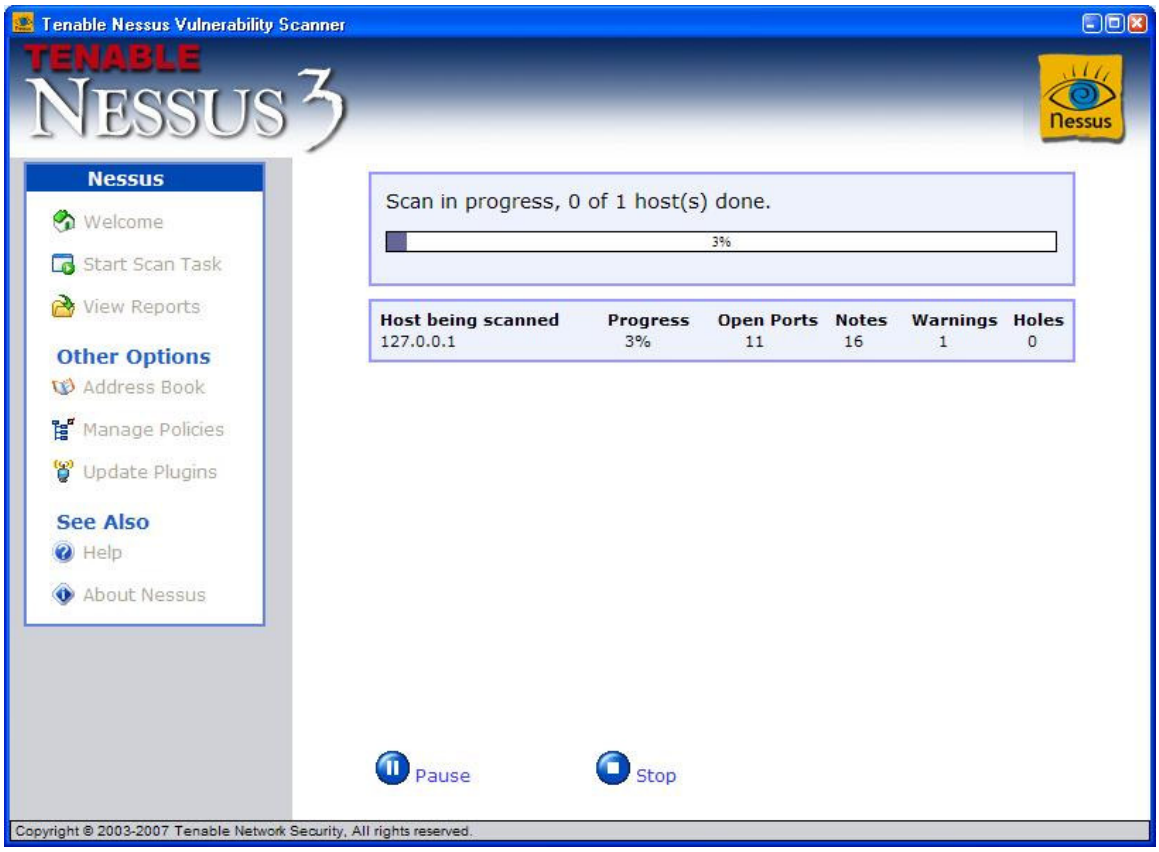


Figure 13: Nessus scan progress

When the scan is complete, Nessus will output a report of what was discovered. This report will also indicate the time Nessus took to complete the scan task. The report is saved within Nessus for future viewing. The content of the reports will be covered in the next section.

View Reports

After each scan task Nessus will make a report of what was discovered. These files will be opened in Internet Explorer (Even if you have another browser such as FireFox set as your default browser). When Nessus is installed the user will have a folder in his Documents and Settings called Tenable. In here are stored the XML and HTML files of the reports. From here we can open with other browsers. Figure 14 provides an example of a report made by Nessus.

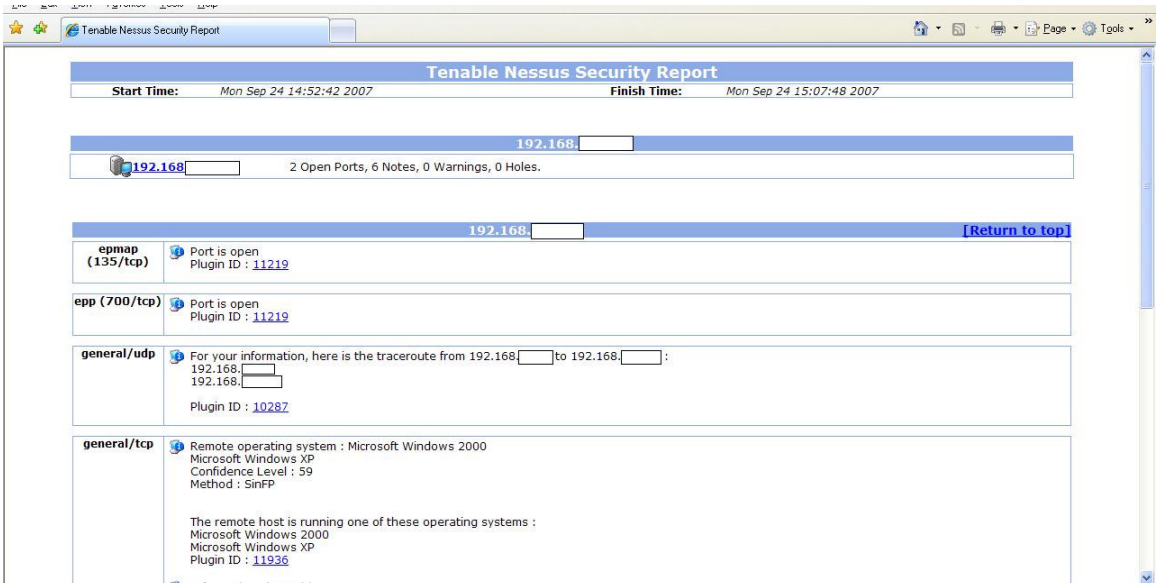


Figure 14: Nessus sample report

The sample report shows one of our controllers on a machine that punches metal called a turret. This PC is NT compatible. The controllers of these machines have a low power PC that runs at 733 Mhz and 128 MB of RAM. From the report we can see that ports 135 and 700 are open. This is typical as these ports need to be open for our software to communicate. The report also shows the IP address of the machine and will do so regardless of whether the IP is static or set using DHCP. For some reason, Nessus had some trouble figuring out what type of operating system is running on the controller. Nessus will usually tell you right away if the OS is Windows 2000 or XP. This may be a result as the OS is not a full Windows OS but a stripped down version. Nessus rates the confidence in the OS. Here we see that the controller has a Confidence Level of 59. This controller is not setup to get on the Internet so the low confidence rating should be due to the OS not having all the Window Updates. The OS for this controller is Windows 2000 Professional. If you click on the Plugin ID number you will be directed to a page describing what the flaw is and where to find information from a CVE on how to prevent the vulnerability. In Figure 15, there is an example of clicking on the plugin and looking at the information of having port 21 open.

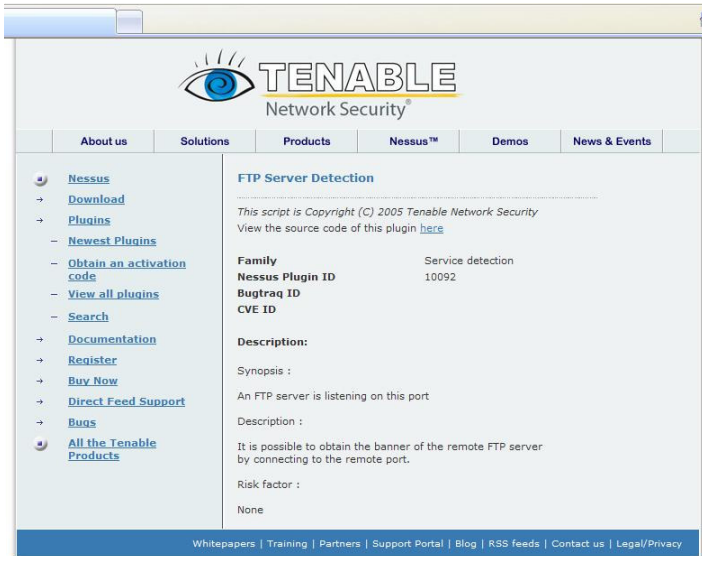


Figure 15: Nessus plugin information

From here we can see the plugin ID, the synopsis, and a description of this plugin. In Figure 16, we have an example of the source code of this plugin that can be reached if you click on View the source code of this plugin here button.

```
#
# The script code starts here
#

include("ftp_func.inc");

port = get_kb_item("Services/ftp");
if (!port) port = 21;

banner = get_ftp_banner(port: port);

if(banner)
{
if("NcFTPd" >< banner)set_kb_item(name:"ftp/ncftpd",
value:TRUE);
if(egrep(pattern:".*icrosoft FTP.*",string:banner))set_kb_item
(name:"ftp/msftpd", value:TRUE);
if(egrep(pattern:".*heck Point Firewall-1 Secure FTP.*",
string:banner))set_kb_item(name:"ftp/fw1ftpd", value:TRUE);
if(egrep(pattern:".*Version wu-.*", string:banner))set_kb_item
(name:"ftp/wuftpd", value:TRUE);
if(egrep(pattern:".*xWorks.*", string:banner))set_kb_item
(name:"ftp/vxftpd", value:TRUE);

report = desc["english"] + '\n\nPlugin output :\n\nThe remote
FTP banner is :\n' + banner;
security_note(port:port, data:report);
}
```

The source code of the plugin above is subject to copyright restrictions. Please check the license of the plugin and/or ask its author for licensing details.

Figure 16: Plugin source code

From the Welcome Menu we can click on View Reports to view all the reports we have made with Nessus. Figure 17 displays this screen.

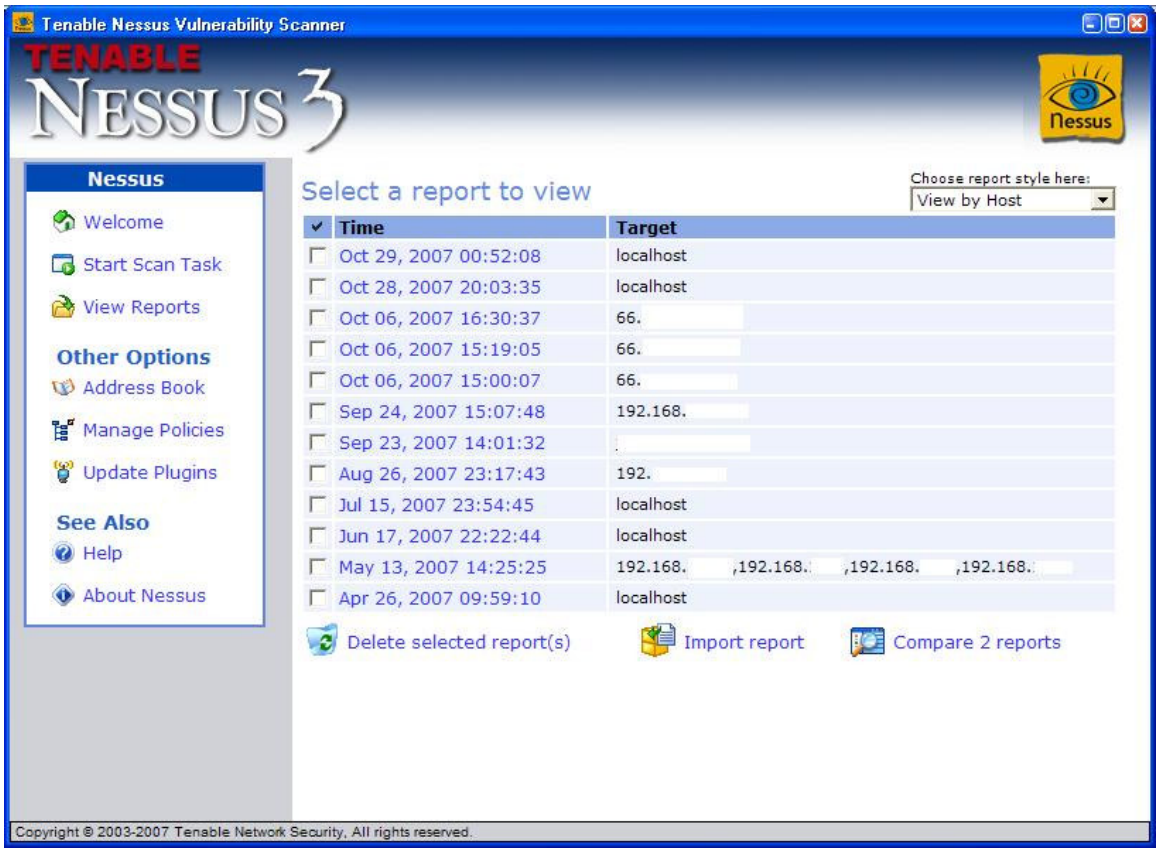


Figure 17: Nessus view reports screen

Nessus gives us the time and date the scan was run and what computers were scanned. From Figure 17, we can see that we scanned the localhost, internal hosts by IP and computer name, and the outside of the network. In the upper right hand corner are options on how to view the reports. The options for viewing the reports include

- View by Host
- View by Port
- View by Vulnerability
- Plain XML
- Plain Text (NSR)

If we do a View by Host, all the hosts will be listed by their IP's or computer names for the user to scan through. If we use the View by Port then we can see the vulnerabilities in the report by the port number and which computers are affected. This style is also used for View by Vulnerability. If we choose to view by Plain XML we will get a XML view of the report. If the user was to make some type of batch or program to scan through these files they could get a desired output that they wanted. If we use Plain Text we can see the report outputted in plain text within Internet Explorer. This, as well, can be scanned by a batch or program created by the administrator.

From this page we can delete any logs that we no longer need by checking in the check box. We can also import other reports that may have been done by another computer. We also have the option to compare two reports. By doing this we can get a report that compares computers and shows us if two of the listed computers are having the same problem and which computers have the unique problem. Figure 18 gives us an example of a comparison report.

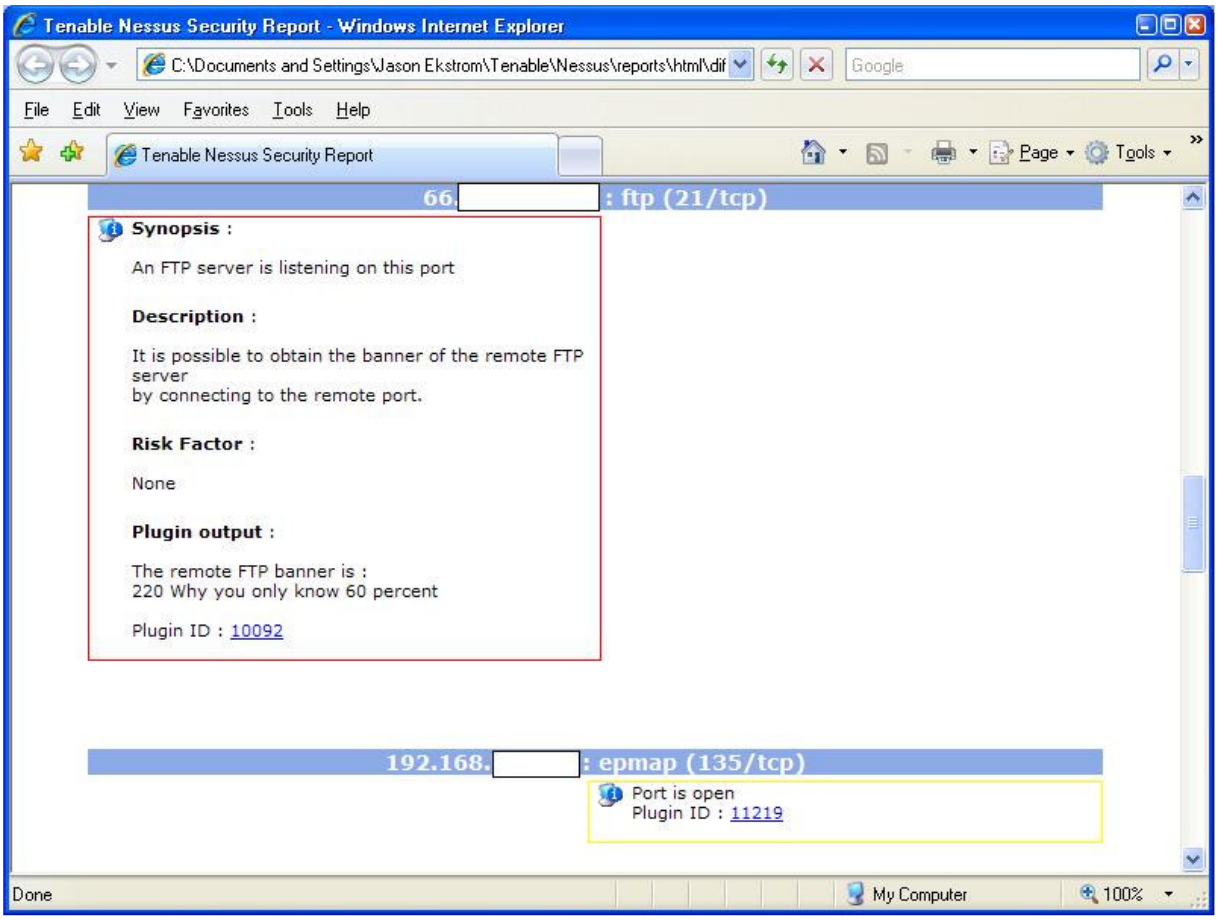


Figure 18: Compare two reports example

From this example, we can see the computer with the IP starting with 66 has an FTP listing on port 21. All of this computers vulnerabilities will be on the left hand side and within a red box. The computer beginning with 192.168 has port 135 open. If 192.168 had an FTP listing on port 21, then there would be a yellow box in the above ports area. The red and yellow box will not be side by side but the red will be on top on the left and the yellow will be below on the right making these reports lengthy. You can only compare two computers at a time. We can apply the different views of reports to this such as by host, port, etc.

Manage Policies

The next option in Nessus is the Manage Policies option, shown in Figure 19.

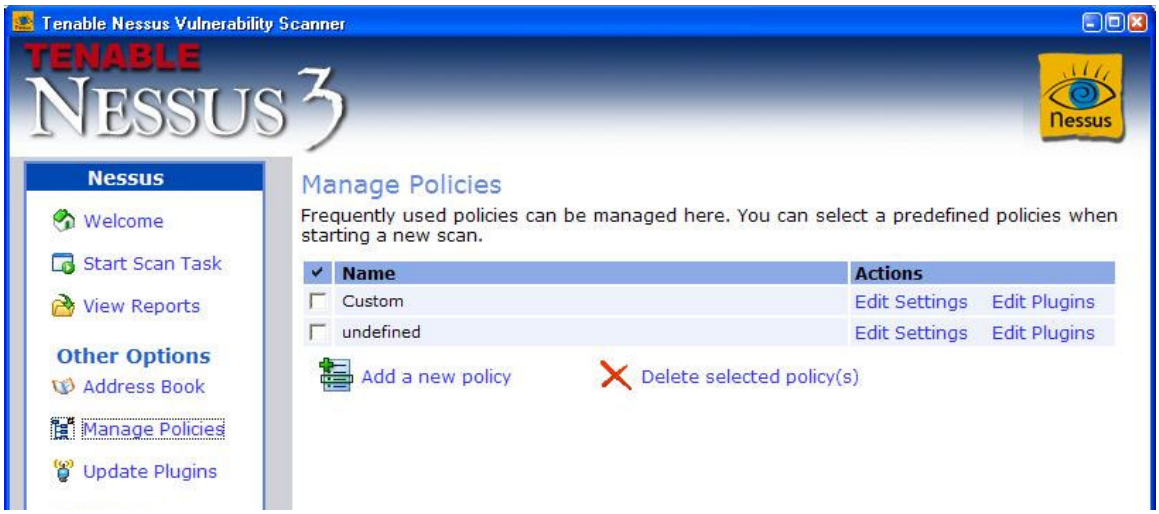


Figure 19: Manage policies screen

In manage policies, we can create policies of the settings and plugins we want to run when we do a scan. By doing this, we can look for vulnerabilities that we are concerned with and avoid having to do a full scan. When going through the scan task wizard, we have the option to pick a predefined policy so we do not have to manually set what settings and plugins we want to run. Figure 20 provides a screen shot of the Edit Settings menu.

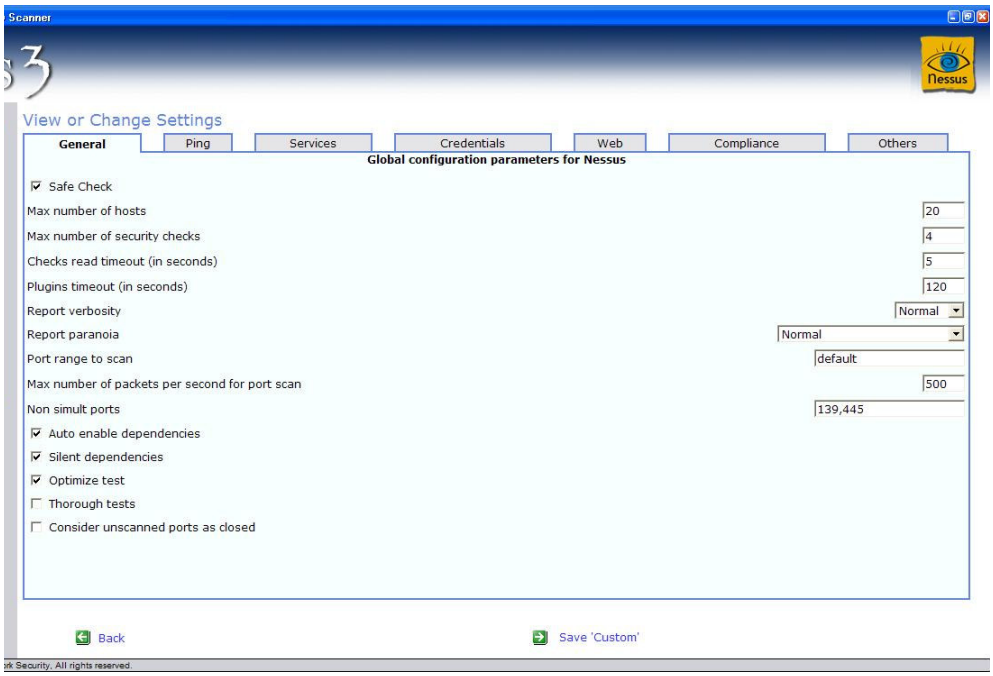


Figure 20: Nessus manage policy settings

The edit settings of the manage policies tool employs 7 categories:

- General
- Ping

- Services
- Credentials
- Web
- Compliance
- Others

With the Edit Settings and Edit Plugins features of Manage Policies there are a lot of options. I will give the general idea of what each of these settings are as there is too much to cover in this paper.

In the general tab of edit settings, we have the option to check or uncheck Safe Check. Ordinarily, run with the safe check, because if this was to provide a DoS attack and the vulnerability was apparent in the computer or router, they will go down. In here we have control over how many hosts we want Nessus to scan. The more hosts, the longer Nessus takes. We can also specify how long Nessus should try the plugin until Nessus should consider the plugin a timeout. We can specify a range of ports to test and which ports we do not want to test at the same time in case this could upset some services operation. We can specify the number of packets to be sent as well so that we do not over burden a particular service.

The next tab is Ping where we describe what types of pings we want to send. We can send ARP, TCP, or ICMP pings. We can modify how many retries the pings can have as well. Figure 21 displays our options for this tab.

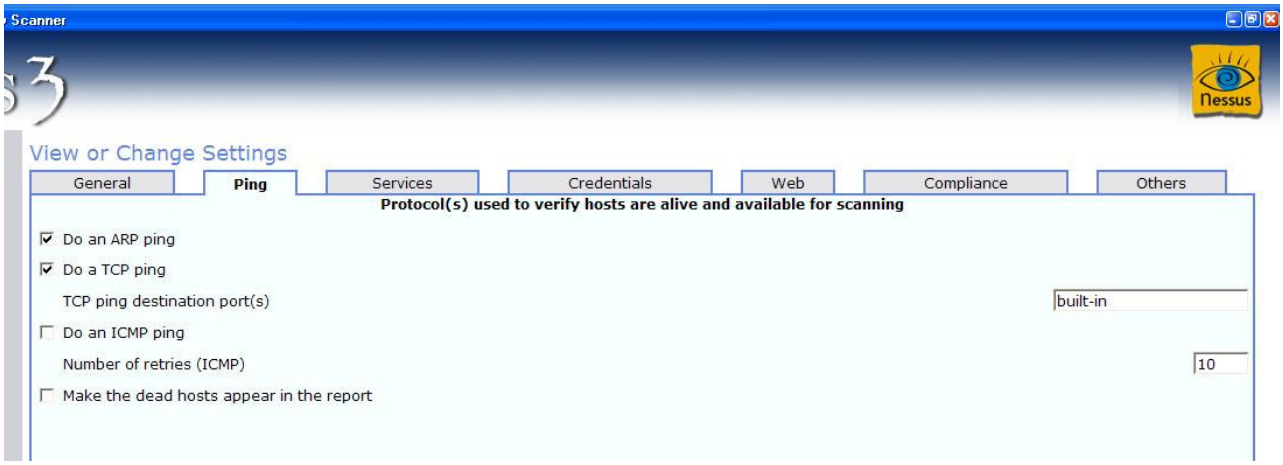


Figure 21: Ping tab of change settings

The services tab is used to specify how many services can be running at the same time. This can help reduce time and could work against you if too many are running and slowing down the Nessus server. We can give a time in seconds as to when Nessus should consider a read/write or network connection a timeout in case there is a problem with that service or if the service is simply not there. We can also test our SSL connections on all ports or known SSL ports. Figure 22 displays our services settings.

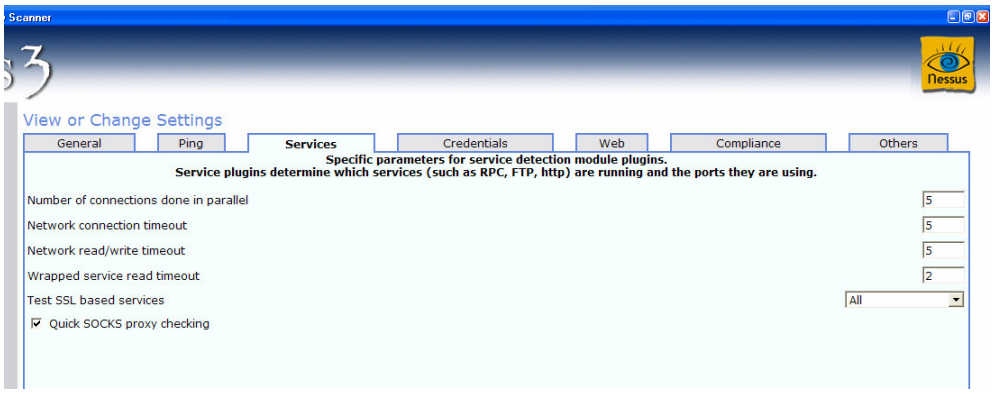


Figure 22: Change settings services tab

If we have a high security network and we use such authentication methods as Kerberos or SSH, Nessus can log into these services and check for vulnerabilities. To do this we need to go into the credentials tab and give the authentication information. Here we can put in the SMB account information, the Kerberos Key Distribution Center, or the SSH username or public key information. Figure 23 displays our options under the credentials tab.

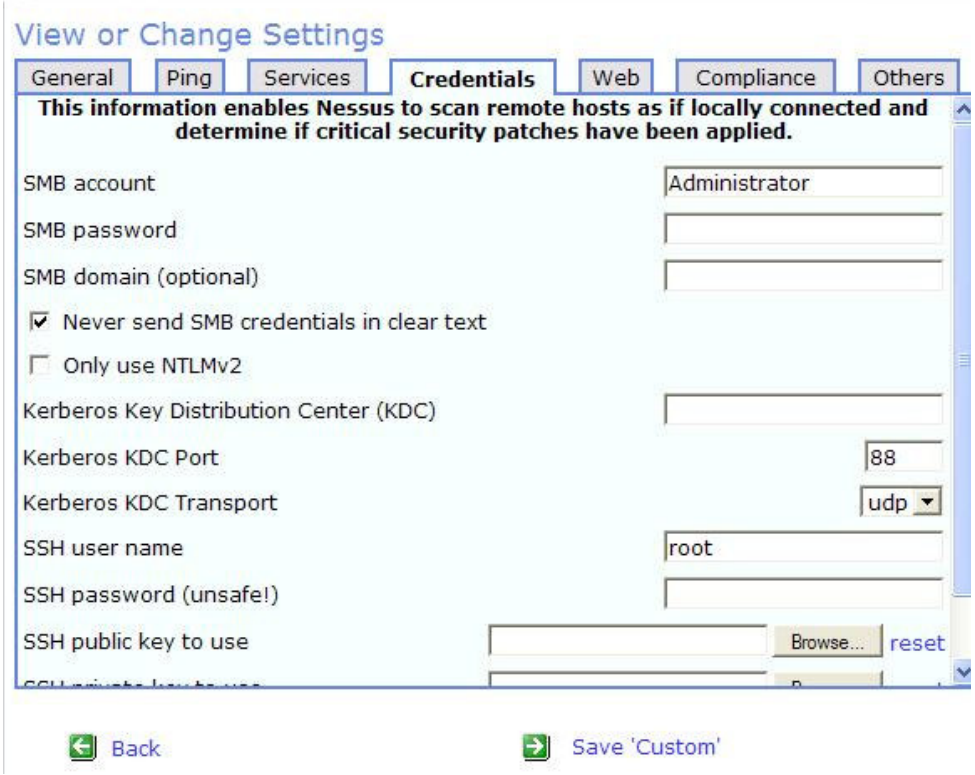


Figure 23: Change settings credentials tab

If we are scanning a web server, then we can make custom settings for authentication to the web server. We can put in the user account and password that Nessus will use to login. This can check for web pages that mirror your page. Figure 24 displays the web tab's options.

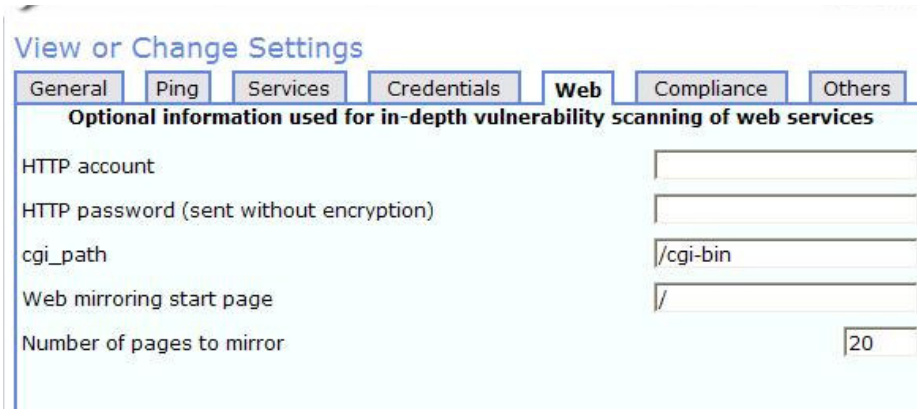


Figure 24: Change settings web tab

The next tab is the compliance tab. This is only used if you purchased a license with Nessus. From here the people at Tenable will make a policy for your network that can be run by Windows or UNIX. You would set where the file is located and Nessus will follow the policies instructions. This is displayed in Figure 25.

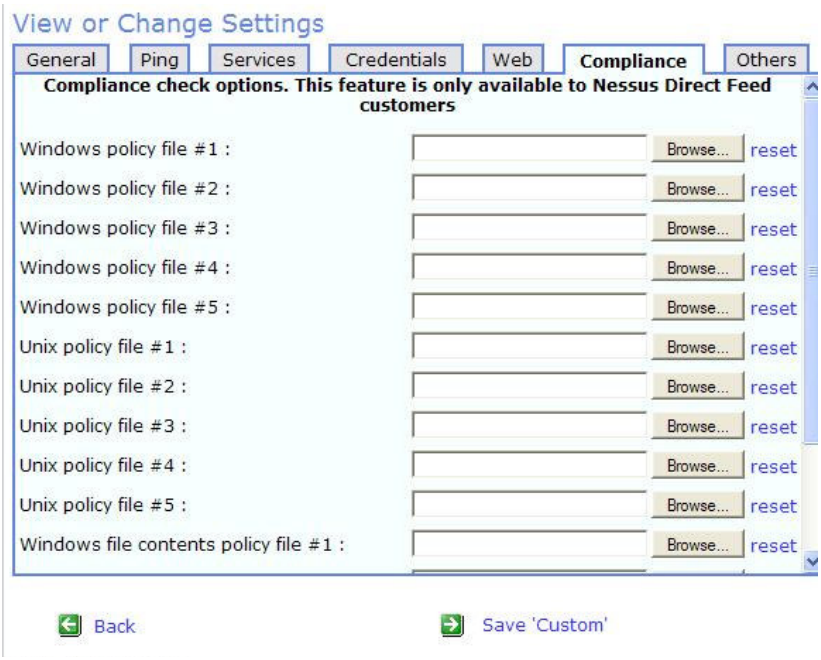


Figure 25: Change settings compliance tab

The final tab is labeled Others. Here we can put in authentication options for servers that have a special purpose. We can first search the computer for such information as user SIDs on the PC. We can make such setups such as if we were testing an email server, we can put in either a SMTP, POP, or IMAP information for Nessus to login with. We can also set Nessus to log into FTP, SNMP, or NNTP servers. Figure 26 displays our options within the others tab.

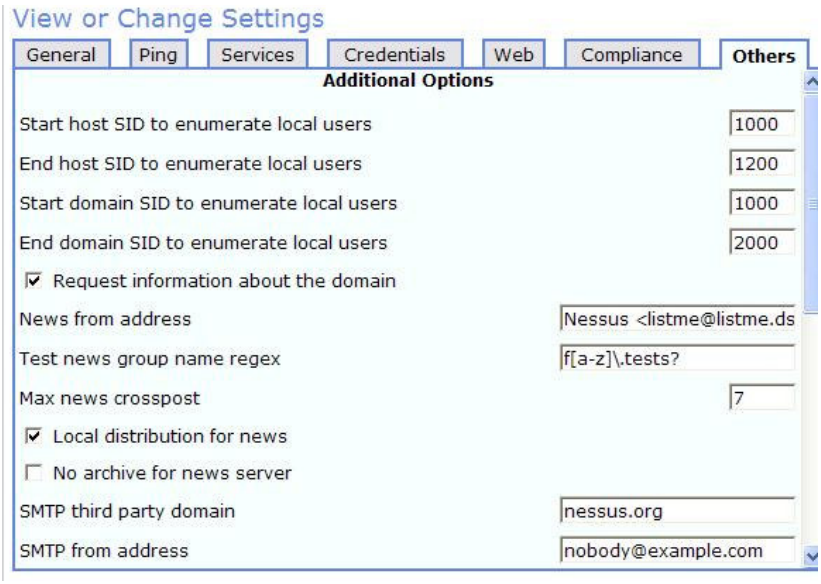


Figure 26: Change settings others tab

The other option we have in the change settings is setting up which plugins we wish to run. Obviously if you are running a Windows environment we do not need to run the UNIX plugins. In Figure 27 we can take a look at some of the plugins and services options we have to run with Nessus.

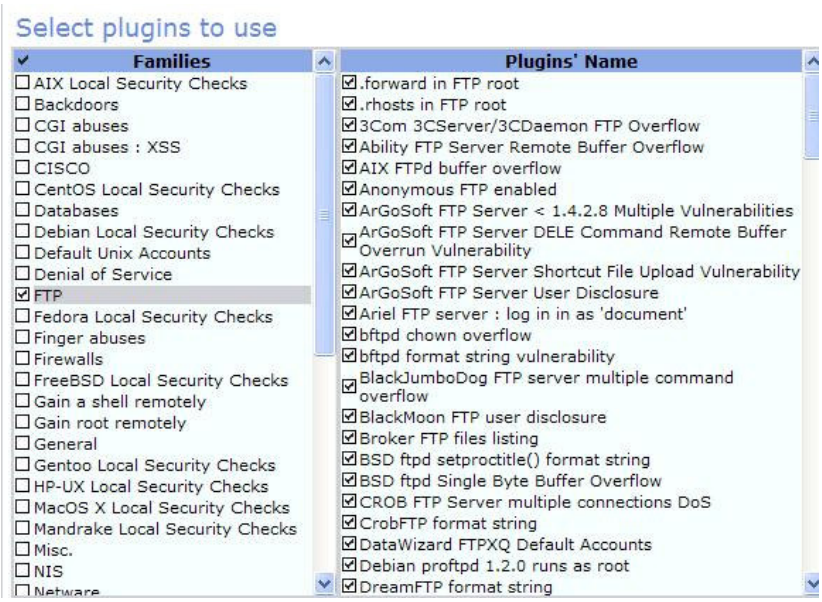


Figure 27: Change settings plugins and options

From Figure 27 we can see the different families of plugins that can be run. We can set plugins for CISCO, Firewalls, etc. When you click on a particular family, you can then see all the plugins that are related. Then you can turn off the plugins that are not a threat or do not apply to your server. I would highly recommend going through this feature. Not only will this save time

in the scans but you do not need to get a lot of false positives in your reports. If you enable all plugins and run on a computer you may get some results that are not related to that OS.

Update Plugins

Finally we come to the Update Plugins feature. As previously mentioned in this paper, if you are using a free license with Nessus, you will get updates a week after they are actually released. When doing an update, you download the entire database each time. The files are compressed in a zipped file and are extracted after they have been downloaded. This takes some time to run as the download, extraction, and installation of the plugin updates takes around 5 minutes. On the update screen Nessus mentions how long you have waited since your last download. Figure 28 displays the update plugins screen.

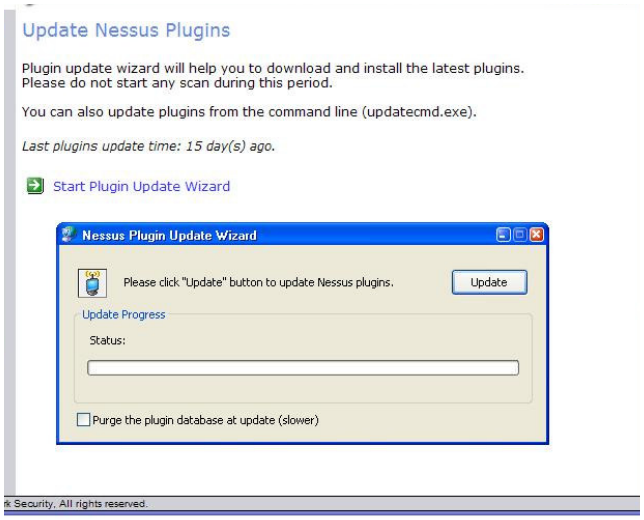


Figure 28: Update Nessus plugins screen

The Experience at Metalsoft

Nessus is a software that works quite well for our needs. The installation and use is very simple. Even though I did not take full advantage of all the options or customizing the software, Nessus has benefited Metalsoft. When first using Nessus we were getting some strange warnings about Linux kernels not being up to date and the computer was running Windows but we figured out how to use the software better and learned a lot about Metalsoft's network. There was some concern when running the software that it would take Metalsoft's router or server down. I had to use Nessus remotely so if there was a problem where the server or router went down, I would have to drive to the sales office in Itasca to get the router or server back up again.

Metalsoft Network Vulnerability Scan

When we first started doing the scans on the network for all host, it would take up to a half hour using the safe check and scanning for all plugins. By changing the settings and setting the

plugins we wanted, we could get the scans down to 15 minutes doing the entire network at once. We did tests on the inside network and I would scan the router from the outside.

Server Scan

The most important computer to scan was the server PC since this is the heart of the network. We setup a custom configuration for the server so that we could scan the services that were running on the Metalsoft server. The plugins we turned on were:

- Backdoors
- Denial of Service
- FTP
- Finger Abuses
- Peer-to-Peer File Sharing
- Port Scanning
- RPC
- Remote File Access
- SNMP
- Service Detection
- Settings
- Useless Services
- Web Servers
- Windows
- Windows: Microsoft Bulletins
- Windows: User Management

Most of the settings were left to the defaults as we felt they were safe. We did set the Others tab to login to the FTP. We allowed Nessus to login as myself as that account is always active. From the first scan we found 28 open ports, 16 warnings, and 2 holes. This was expected, since there are a lot of services running on the server. We began to look into each issue one by one and try to figure out what Metalsoft could do in order to make the server as secure as possible. This was a long and tedious process that needed a lot of research. There are still some issues that need to be resolved but we addressed the major issues that needed to be taken care of.

From the scan there was a confidence level of 99 meaning that all updates were in place for Windows. The server is set to do automatic updates and anti-virus updates daily. This was a relief that these services were running properly and can run without user intervention.

I will discuss one major issue we found during this effort. Nessus had labeled this as a hole in the server. Since the server is not used as a web server very often and was not currently public, we were not too concerned with this since the service is always off. We needed to address this situation as we do not know when we will need to get the web server up and running. Figure 29 displays what Nessus had found with Metalsoft's IIS.

http (80/tcp)	The use of /iisadmin is not limited to the loopback address. Anyone can use it to reconfigure your web server. **Solution**: Restrict access to /iisadmin through the IIS ISM **Risk Factor** : High CVE : CVE-1999-1538 BID : 189 Other references : OSVDB:273 Plugin ID : 10358

Figure 29: Hole in iisadmin

From Figure 29, we can see that the iisadmin is not limited to the loopback address. I checked the plugin ID and the Bugtraq ID number 189. This took me to http://www.securityfocus.com/bid/189/info. This explained to me that there is a DLL installed with IIS called ISM.dll. ISM stands for Internet Services Manager. This was for remote administration of a web page. (Litchfield, 1999, Discussion, para. 1) Metalsoft does not use remote administration for the web page so the service is not needed. An example of how to exploit this would be to go to http://www.somewebpage.com/scripts/iisadmin/ism.dll?http/dir. Whoever goes to this page will see a prompt for a username and password. If the hacker gets in they may not be able to make changes to the server but can gain sensitive information about the web server and its configuration. The fix for this was to go to C:\WINNT\system32\inetsrv\iisadmin and delete the ISM.dll.

Along the way, there were a few simple administration changes to make to clear out some vulnerabilities. There was a port open for remote administration of the FTP. Since we do not use remote administration for the FTP the port was shutdown. At one point there was a Metalsoft employee who logged into the server with Terminal Services to check sales leads. An account was created for them and Nessus had found that their password was the same as their username. When the account was first created the password was not the same as their username so they must have changed the password themselves. The employee no longer needs access to this server so the account was removed.

There are a few issues that we are looking into on the server. We seem to have some flaws in the DNS and some ports are listing as unknown. We are continuing on investigating these issues but they do not seem to be of big concern.

Router

When first testing the router, I scanned from the outside at home. At first we were using Nessus version 3.0.5. The safe check option was turned on and the scan had the dangerous plugin option turned on. Nessus discovered that the router could be crashed by a DoS attack. The router did go down and I had to drive to Itasca and restart the router. After upgrading to Nessus 3.0.6 we could do the same scan and the router would not crash. This vulnerability is still open but there has not been an update to fix the problem. Some other information Nessus found from the scan was that the services that were port forwarded, such as the FTP and Terminal Service, could be seen from the outside.

While scanning internally, Nessus found more flaws in the router. The information is sensitive and can not mention in a public paper. There are few warnings, but the reported fixes generally involve contacting the vendor.

Salesmen Computers and Printers

Nessus did not find a lot of problems with the salesmen computers'. They typically had 3-6 ports open, 0 holes, and 6 warnings. The scans were directed to finding Windows flaws. The salesmen do not have any special services running on their PCs such as SQL. The salesmen had received new laptops about 5 months ago. Apparently, they have been doing a good job on keeping the computers clean and updated.

Some things that were strange was I had all the plugins enabled and all of their computers were vulnerable to a DoS attack that affects computers running Linux kernel 2.6. I was advised to update them to kernel 2.6.7 or higher but they are all running Windows XP SP2. Their vulnerabilities only seemed to affect them by attacks such as DoS. At least the salesmen did not have unexplained ports open and no Trojans or viruses were discovered.

On the other hand, the printers do have some problems. From the Nessus scans, we found that the printers have some problems that are known and do not have a solution. Figure 30 reports these problems.

The lpd daemon installed on the remote host appears to be from the Hummingbird Connectivity suite and to suffer from a buffer overflow vulnerability. An attacker can crash the daemon by sending commands with overly-long queue names and, with a specially-crafted packet, even execute code remotely within the context of the affected service.

Solution: None at this time.

Risk Factor : High
CVE : CVE-2005-1815
BID : 13788
Plugin ID : 18403

A vulnerability in the NIPrint could allow an attacker to remotely overflow an internal buffer which could allow code execution.

Solution None, Contact the vendor
http://www.networkinstruments.com/products/niprint.html

Risk Factor : High
CVE : CVE-2003-1141
BID : 8968
Plugin ID : 11926

Figure 30: Printer vulnerabilities

Both printers are made by Hewlett Packard and are connected to a HP JetDirect to be network printers. Both printers are subject to buffer overflow. If I click on the link that is under solution I go to NIprint's web page and I see that they have discontinued making and supporting this service back in September of 2006. (Network Instruments, LLC, 2006, NIPRSUP1013) Since this is a small network with no visitors who come on the network, this is not a big concern.

Some of the other warnings about the printers is that it says that SNMP and FTP services are running on them but I have not had time to go over there and check this out.

Other Experiences with Nessus

I perform software support for Metalsoft Inc. I was working with a customer who kept getting these strange errors from a product we produce. Figure 31 displays the picture that the customer sent me.

Figure 31: Customer error messages

I had never seen this before. The customer had two computers with the software installed. This occur every couple of weeks. I checked the computer out, reinstalled the software and the problem was not resolved. These messages do not tell me anything and Windows did not log any problems. When they would occur there were always five of them in a row. I was using Nessus on my computer on which I have the same software installed and all of a sudden these messages started to show up. Apparently, the customer's IT department was using a vulnerability scanner on the network and causing these errors. From the report I can see the five ports the software had trouble with. They are displayed in Figure 32.

epp (700/tcp)	Port is open Plugin ID : 11219
apache-administration-server (8089/tcp)	Port is open Plugin ID : 11219
unknown (8085/tcp)	Port is open Plugin ID : 11219
sunproxyadmin (8081/tcp)	Port is open Plugin ID : 11219
unknown (8084/tcp)	Port is open Plugin ID : 11219

Figure 32: Software open ports

All of these ports were affected by plugin ID 11219. What this plugin does is send SYN packets between the Nessus server and the host computer rapidly. The software does not like this. This does not cause any problems in the software. All the services remain running and connection is intact. I tried to bring this to our developers' attention but this product is no longer supported in the US so they were not going to investigate this matter any further if there was a security vulnerability.

My office is located in Mount Prospect, IL. The network at this office is maintained from a staff out in California. Apparently they do have an IDS system installed. IDS stands for Intrusion Detection System and this can be software or hardware that takes action if something were to occur on the network. I am not apart of the IT department at this office so I would not be informed of what is in place. Since there are no IT people onsite, a lot of the network issues are directed towards me. I got an email saying that a computer on the network was sending too much traffic. All they gave me was the IP address. I had no idea what computer it could be. I opened Nessus and put in the IP. Nessus does a good job of finding host names and what domains or workgroups computers are in. I started the scan and when I was at 10 percent of the scan I was shut off by their IDS and the scan completely stopped. I got a report saying that the computer was either shutdown or the connection was terminated by an outside resource. Their IDS had terminated my connection to the computer. Nessus did not tell me what software or hardware may have made the action but at least I know there is something in place.

Nessus Review

Nessus is extremely useful. I discovered a lot on the Metalsoft Itasca site's network. Since this was a small network, I could just scan each PC individually. The resulting reports provided me an easy way to view problems. Scrolling through a report that had seven nodes was difficult. I would imagine this could be difficult to implement on a large network of over 100 users. Administrators would need to have some sort of program to filter through these reports. The software is straightforward and I did not need to refer to documentation too often. I am sure if I am to make my own Nasls that I would need to see more documentation and would need to find tools to test if the Nasls were working.

WireShark

Wireshark is a GUI network protocol analyzer that captures packets over the network and stores them to file. Wireshark can be used on and offline. The current version of Wireshark is 0.99.6a and is the replacement for Ethereal. The project was first started in 1998. Wireshark is an award-winning network protocol analyzer that can be used on such platforms as OS X, Linux, UNIX, and Windows. The software is free to download and is considered the defacto standard across many industries and educational institutions. Some of the features that are included with Wireshark are:

- Hundreds of protocols supported
- Live capture and offline analysis
- Standard three-pane packet browser
- Multi-platforms
- Network data viewed through a GUI
- Powerful display filters
- VoIP analysis
- Read and Write in different capture formats
- Capture File Compression
- Live data can be read from Ethernet and other network systems
- Decryption support for many protocols
- Coloring rules can be applied for analysis
- Output can be exported to XML, PostScript, CSV, or plain text.

Wireshark can be used to troubleshoot network problems, examine security problems, debug protocol implementations, and learn network protocol internals. Wireshark contains a lot of features and is open source. For the purpose of this paper, I will explain why and how I used Wireshark, as there are so many possiblities for using the software. The purpose for using Wireshark for Metalsoft was to see what network traffic was like on the network. We needed to know if there was bad traffic and what was the norm. The FTP logs displayed that people were trying to enter Metalsoft's FTP and Metalsoft needed to know what they were trying to do. Since Metalsoft had other remote services running we needed to check if people from the outside were trying those services as well.

Prerequisites

Since I installed Wireshark to a Windows environment I will discuss what is needed to run Wireshark in Windows. The system requirements for Windows are:

- Windows 2000, XP, 2003, or Vista
- A 32-bit Pentium processor of 400 MHz or greater
- 128 MB of RAM
- 75 MB of available hard drive space
- 800x600 resolution with 16 bit color
- A supported network card (Any card supported by Windows should do)

Metalsoft's server hardware is:

- Pentium IV processor with 2.8 Ghz
- 1 GB of RAM
- 30 GB free Hard Drive Space
- 1280x800 resolution with 256 bit color
- Supported Windows network card

Deployment

You want to put the Wireshark host somewhere on the network where the traffic passes through. Users should avoid using Wireshark in a virtual machine as there can be confusion between the host and the virtual machine's packet data. Wireshark was installed on Metalsoft's server since this PC controls the DNS, FTP, Terminal Services, and just about all the traffic that travels on the network.

Installation

To obtain a copy of Wireshark you can go to http://www.wireshark.org and go to their download page. You will be able to download executables or binaries if you wish to customize your Wireshark. From the download page you will be directed to mirror sites that contain Wireshark. Wireshark is a free software under the GNU GPL licensing. The installation of Wireshark on Windows is simple.

First you need to run the setup.exe so that the Welcome screen appears and click on next as show in Figure 33.

Figure 33: Wireshark welcome screen

Agree to the GNU GPL license agreement as shown in Figure 34.

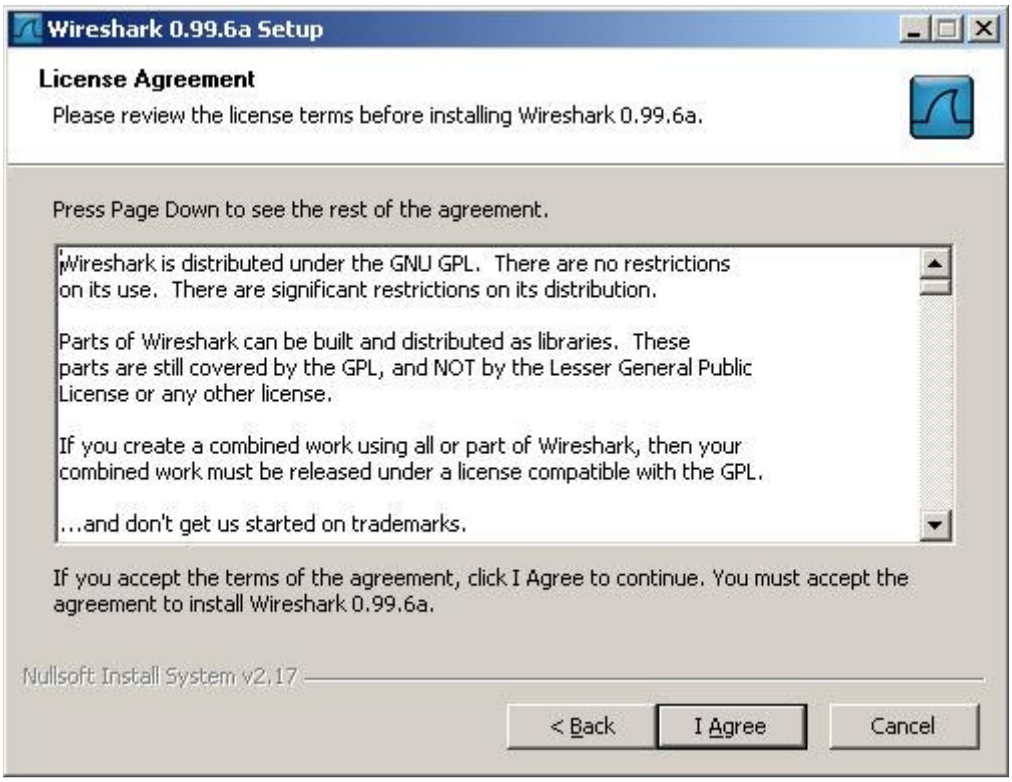


Figure 34: GNU GPL license agreement

The next page will ask what components and tools you want to install. The first selection you have is what type of interface you want to install. You have three options. The first is GTK1 which is the old GUI network protocol analyzer toolkit. The second is the GTK2, which uses the modern GUI toolkit. The third is the GTK MS Windows Engine that has a Win32 look. For my purposes I used GTK2 so I can display all the colors for analyze and statistics features. I installed these options on my PC, which I use to analyze the data. Since I log into the server remotely I need to use GTK1 or I will not be able to see the packets that were collected due to the display of Terminal Services. Over Terminal Services the packets will be white and you can only read the information if you highlight the packet. When typing in your configurations you will not see what you are typing and have to trust that the input is correct.

You will have the option of installing the Tshark tool. Tshark is a command-line based network protocol analyzer that uses the same options as Wireshark. This is what I used on the server and will cover in more detail when I talk about my procedure for running Wireshark.

On this page you can select what plugins or extensions you want to install. The first option is the dissector plugin that offers some extended dissections. Then there is the tree statistics plugin with extended statistics. Wireshark offers an experimental display filter engine called Mate. Mate stands for Meta Analysis and Tracing Engine. You also have the SNMP MIB extension for

more detailed SNMP dissection. SNMP stands for Simple Network Management Protocol. This is used in network management systems to monitor the network and bring any problems to the attention of the system administrators.

Then you have four tools that can be installed with Wireshark. Editcap is a program that reads a capture file and writes some or all of the packets to another capture file. Text2Pcap is a program that reads in an ASCII hex dump and writes the data into a libpcap-style file. Mergecap is a program that combines multiple saved capture files into a single output file. Capinfos is a program that provides information on capture files. All of these files can be used with the command prompt and have their own options. Programmers can create custom programs to use these files to output results that are needed.

Last, you can select whether or not you want to install the User's Guide. I would highly recommend installing this. I installed all of these options except for the Mate since it is experimental. All these options can be seen in Figure 35. When you have finished selecting what type of interface and plugins you want to install you can click on the next button to proceed with the installation.

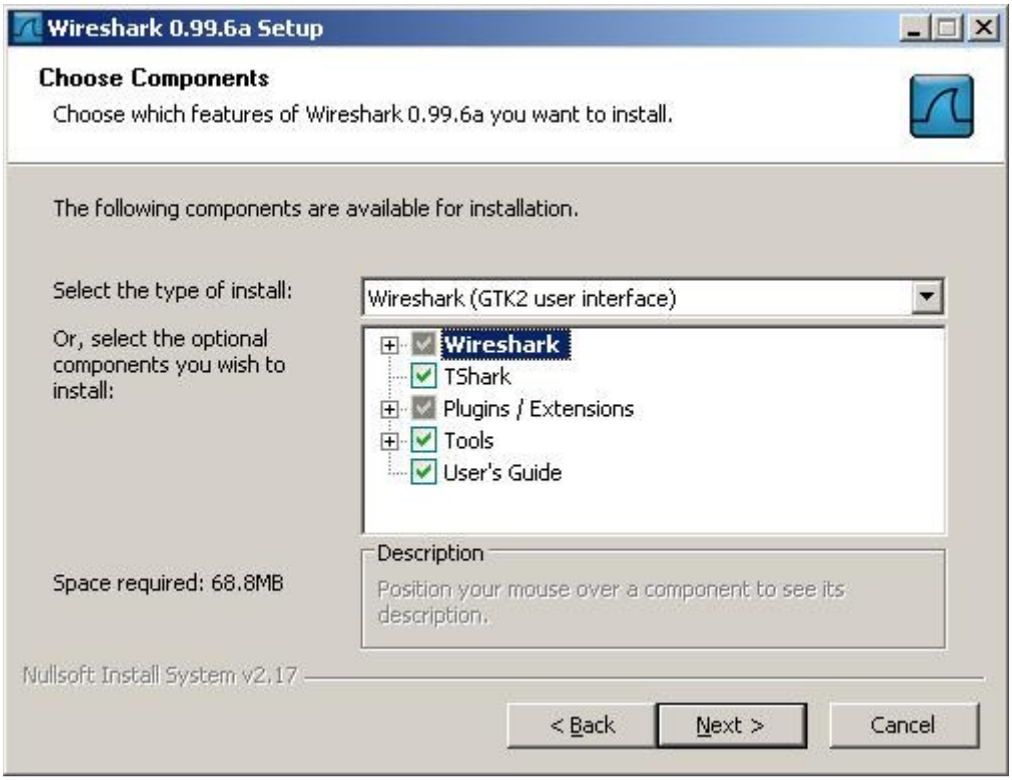


Figure 35: Wireshark tools and extensions

The next screen asks what shortcuts of Wireshark you would like and what type of file extensions you want associated with Wireshark as seen in Figure 36. You can associate Wireshark with such extensions as cap, enc, etc. These are captured packet formats that are similar to what TCPdump or Windump create.

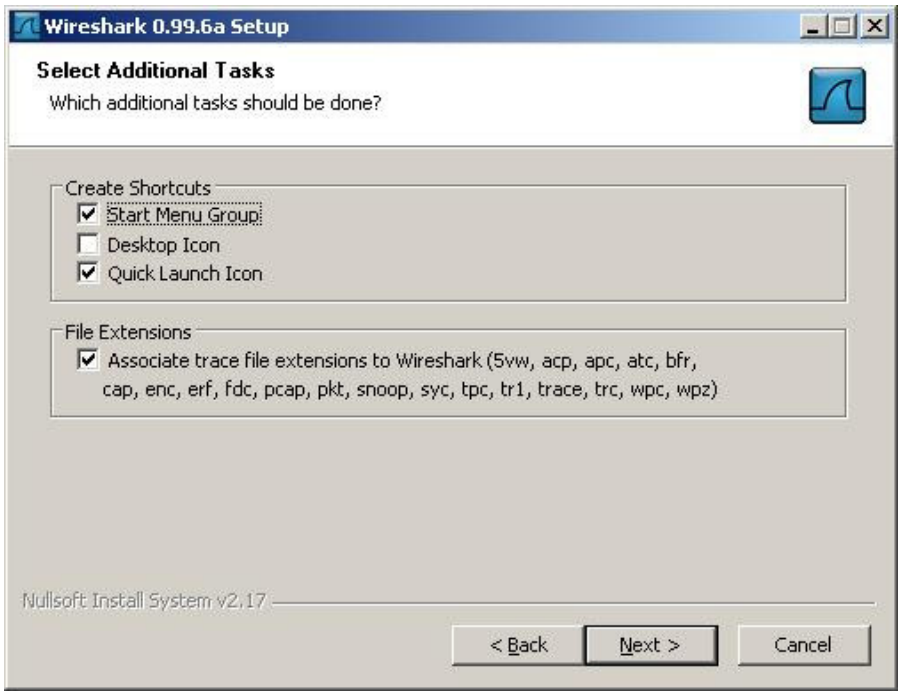


Figure 36: Wireshark shorcuts and file extensions

Next you have the install directory. This is displayed in Figure 37.

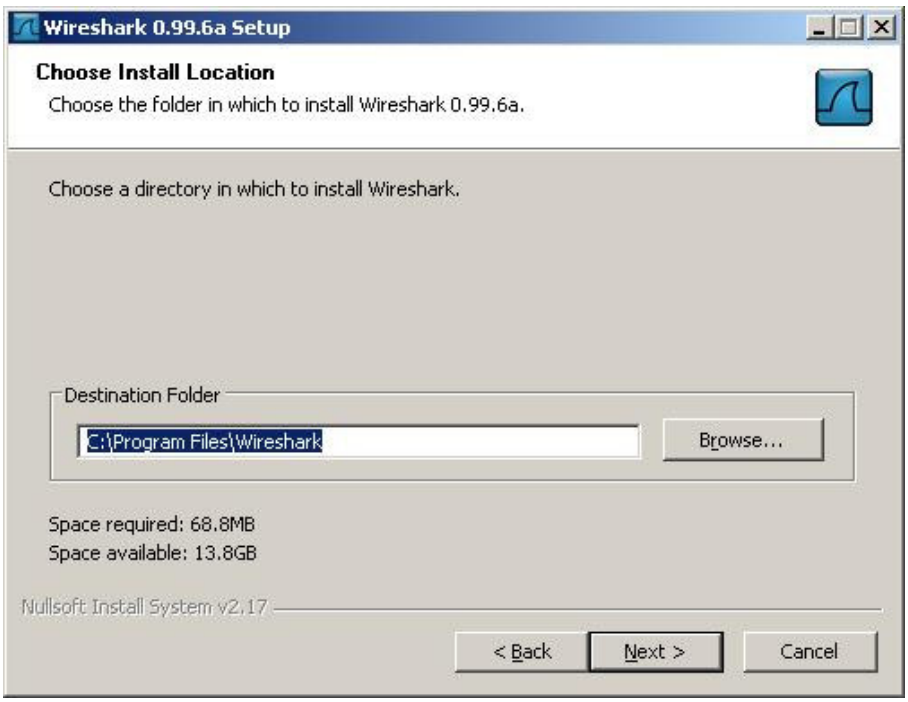


Figure 37: Wireshark install directory

In order to install Wireshark you need to have WinPcap installed. WinPcap is the Windows version of libpcap that is used for UNIX systems. WinPcap stands for Windows Packet Capture and this is a driver for capturing packets. Libpcap stands for Capturing Packet Library. After the install directory page, the installation will search if you have WinPcap installed. If you do not have a WinPcap installed, Wireshark will install it for you. With this version of Wireshark it will install WinPcap 4.0.1. The latest release is 4.0.2 and can be obtained from http://www.winpcap.org. On this page you have the option to check if you want the NPF service to start at startup. NPF allows users without Administrator privileges to capture packets on the PC. NPF stands for Non-Plug and Play driver and is apart of the WinPcap driver. This can be managed in the device manager under hidden devices. This can be started at startup or can be started manually by an administrator. I leave this option off since we do not need to have people watching our network that should not be. After you have decided what you want to install you can click on Install. This page can be viewed in Figure 38.

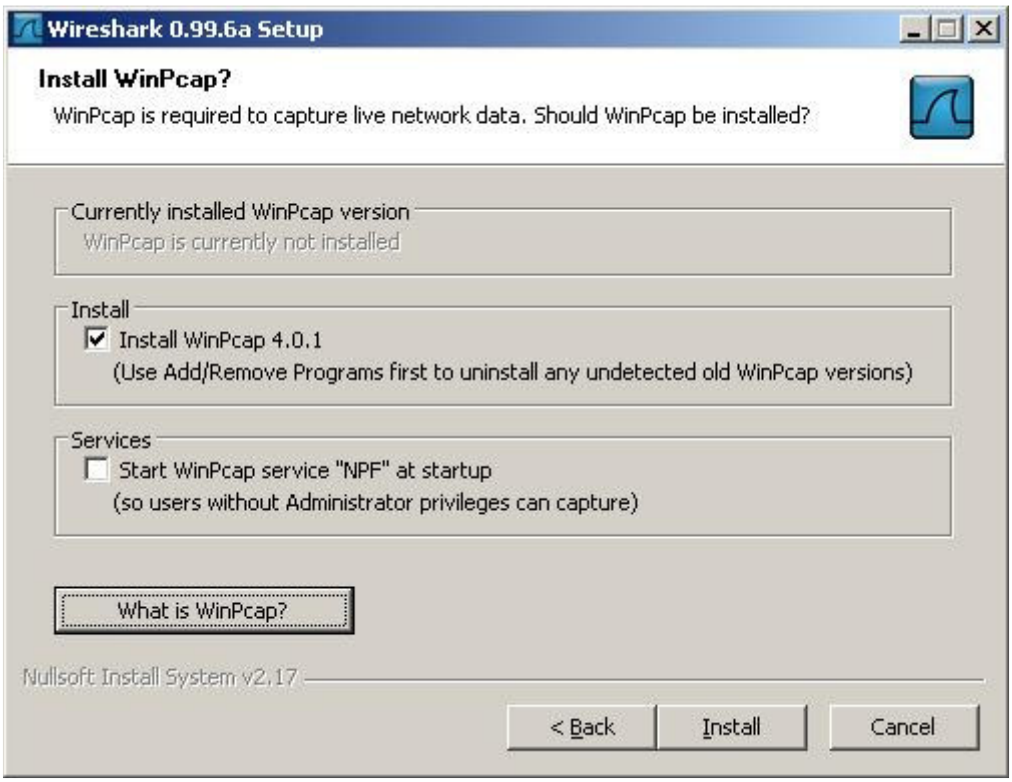


Figure 38: WinPcap installation page

Once you click Install, Wireshark will begin extracting its files. If you do not have WinPcap installed a wizard will pop-up for its installation. WinPcap will only install version 4.0.1 from Wireshark and does not check for updates. To upgrade a WinPcap you need to remove the previous version and install the latest version. You will not need to reinstall Wireshark later if you need to update WinPcap. All you need to do for WinPcap is agree to the license agreement and it will install. Once that is complete you should see a finish page for Wireshark that asks if you want to run Wireshark or to see the release notes for Wireshark 0.99.6. This is shown in Figure 39.

Figure 39: Wireshark finish installation

If you leave the Show News option checked after completion of the installation it will take you to the Wireshark.org web site where you can get the latest information about Wireshark.

Wireshark Procedure

Since Metalsoft was concerned with what traffic was coming in and out of the server Wireshark was installed to the server. To get this portion of the implementation off we had some experimenting to do. First we tried running Wireshark as the GUI all day long. This was a mistake. While Wireshark is collecting packets the output file can become quite large in size. The bigger the output file the more memory Wireshark will use of the computer. According to Joshua Davis of Wireshark, this is a known bug in Wireshark. Wireshark uses memory to store packet data such as conversation and fragmentation data to display on the screen. The memory usage is dependent on the number of packets, the kind of packets, and the preference settings on viewing the packet. The GUI toolkit keeps packets in memory for dissecting the packets information in the packet list for advanced protocol analysis, such as sequence number analysis of TCP. In order to fix this the developers of Wireshark will need to use a new GUI toolkit. (Davis, 2007, KnownBugs – OutOfMemory) Figure 40 shows a screen shot Task Manager opening a Wireshark log. The computer uses 350 MB of memory when sitting idle. With Wireshark opening a 150 MB log the memory jumps up to 707 MB of memory being used and levels there.

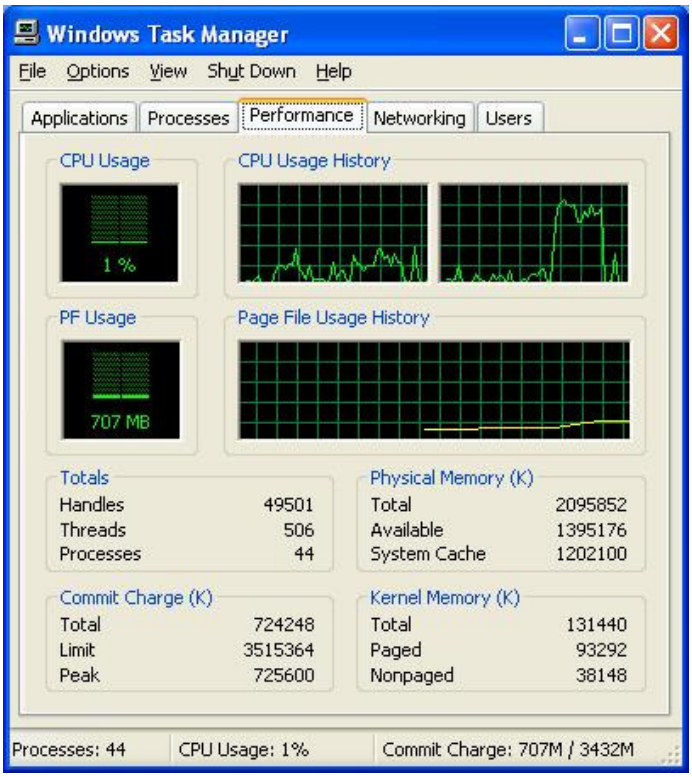


Figure 40: Wireshark task manager memory usage

As Wireshark ran in this configuration, the server had some problems, and the salesmen could not access the server or the Internet. We had to stop the Wireshark from running and reboot the server. The server claimed that it was running low on virtual memory. Naturally, Wireshark was using up all of the system's RAM. After learning this, we took a look into what Tshark could do for Metalsoft. Since Tshark is a command-line based network protocol analyzer, Tshark does not use up as much memory as Wireshark with the GUI open. After consulting the FTP logs of who was trying to access the server we noticed they would typically try to gain access between 12 to 5 a.m. CST. The salesmen work from 8 to 5 p.m. so we did not want to invade their privacy by logging packets that could be theirs. Since the salesmen produce a lot of traffic and the more traffic Tshark collects the more memory is used on the server, we set Tshark to do data collection when the salesmen had left for the day. To accomplish this we created a batch file. The batch file looked like this:

```
cd c:\program files\wireshark
tshark –a duration:43200 –b filesize:102400 –I 2 –N nmtC –t ad –w
d:\users\jasone\Metalsoft_Night_Batch
```

We then used Windows Scheduled Tasks to run this batch process everyday at 6 p.m. The executable for Tshark is located in C:\program files\wireshark. To launch Tshark you can use the command prompt and navigate to the executable's location. Tshark is the command to launch the program and users can put what options they want Tshark to run. An example of Tshark running is in Figure 41.

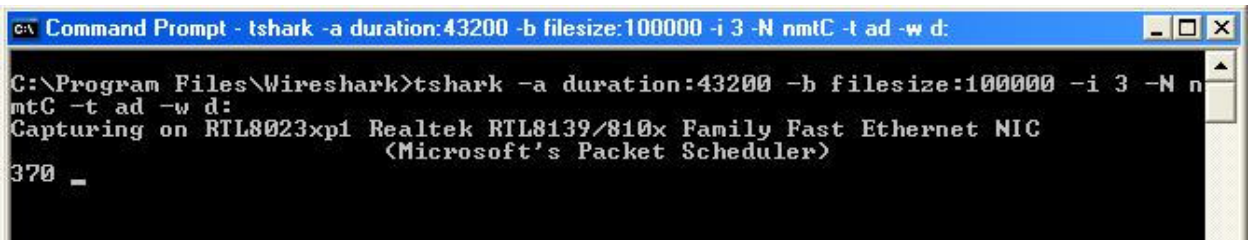


Figure 41: Tshark running

With the batch file created this way, anyone walking by the server would not see Tshark running. Tshark would be running in the background quietly without disturbance from user intervention. Since there are quite a few options to Tshark / Wireshark I will address the options that were used for the assessment.

-a duration:43200

The first option that comes in the command is –a duration:43200. The –a option specifies when Tshark should stop writing to file. There are three options that can be used for this. There is filesize:value. In this option we can specify when Tshark should stop writing to file by saying how big the output file can be. This can be used to prevent the problem where the file got too big and caused the server not to run properly. Value is measured in kilobytes so if you wanted to stop a file capture after the file has reached 100 MB you would put in 102400 for value. This is good for when you are reviewing the logs of captures. A 1 GB file can be verbose and difficult to find what you are looking for. The logs can be filtered by protocol, MAC address, or expression but the search can take awhile to search through a 1 GB file. A second option you have is files:value. In this option you can specify Tshark to stop capturing files after so many files have been written. So, if you want to stop capturing after 5 output files were created you would put in 5 for value. Depending on the other options you are using with Tshark this could be a good control variable for easy viewing of the logs when the scan is done. Metalsoft used the duration:value option. Here we can specify after how much time Tshark should run before it stops capturing. Value is measured in seconds and Tshark would run for 12 hours since we were capturing between 6 p.m. to 6 a.m. when no one was in the office. 12 hours is equal to 43,200 seconds.

-b filesize:102400

The next option is –b filesize:102400. The –b option causes Tshark to run multiple files. The –b has similar options to –a. There is duration:value where you can specify how long Tshark should run before creating a new output file. If you are checking for when batches should run on the network this could be good to use to see if they happen during the hours you wanted them to run. Here we used the filesize:value option. We did not want the log files to get big where the server could not handle them so we a set value of 102400, which is 100 MB. With a 100 MB file the server seemed to run properly and there were not too many files created for each batch. In one night we would typically have one file.

-I 2

Wireshark and Tshark can run on multiple interfaces. The –I option is used to select which interface you want to capture packets from. The server's network card was located on Interface 2. Interface 1 was a dialup adapter for VPN capture. To find what interfaces number is associated with which network adapter you can run tshark –D to view a list of interface adapters.

-N nmtC

The –N option allows name resolultion of the packets as they are captured. Since Metalsoft needed to know as much information as possible all options were turned on. The m enables the MAC address to be resolved. The n enables the network address to be resolved. The t enables the transport-layer port number in resolution. The C enables concurrent DNS lookups on the network.

-t ad

The –t option is for setting the packet timestamp. There are several options that can be used with this. The first is the a option, which is used to display the absolute time the packet was captured without showing the date. The r option is used to display the relative time the packet was captured. This is measured from when the scan was started and is measured in seconds. This is the default option Wireshark uses as timestamp. The d option displays the delta time, which measures the time from the previous packet to the current packet was captured. The e option is used to measure the time the packet was captured since epoch, which is from January 1, 1970 00:00:00. Metalsoft used the ad option. This allowed us to get the absolute time with the date displayed. This option was used to compare the FTP logs time with the corresponding Tshark time.

-w d:\users\jasone\Metalsoft_Night_Batch

The –w option tells Tshark to write to file and where. The Tshark files were saved on partition D in my users folder. The file was given a name of Metalsoft_Night_Batch. When using this option the output file would come up with a filename of Metalsoft_Night_Batch_00001_20071103180006. In the file name the date is displayed in a YYYY/MM/DD format. The number 00001 is created in case during the scan Tshark needed to make another file. If Tshark was running a batch on November 3, 2007 and the file size became large than 100 MB, then Tshark would create another file with the name of Metalsoft_Night_Batch_00002_20071103180006 so that files would not be overwritten.

Reviewing Results

We began running Wireshark in April 2007. To come up with the current procedure that is currently being used for Tshark took a month. Starting in May 2007, we started researching the network traffic to establish what was normal. Tshark would scan everyday and I would review the logs over the weekend. The logs were put into my users directory and I would download them from the FTP. I have Wireshark installed on my PC and would open the logs. An example of a log is in Figure 42.

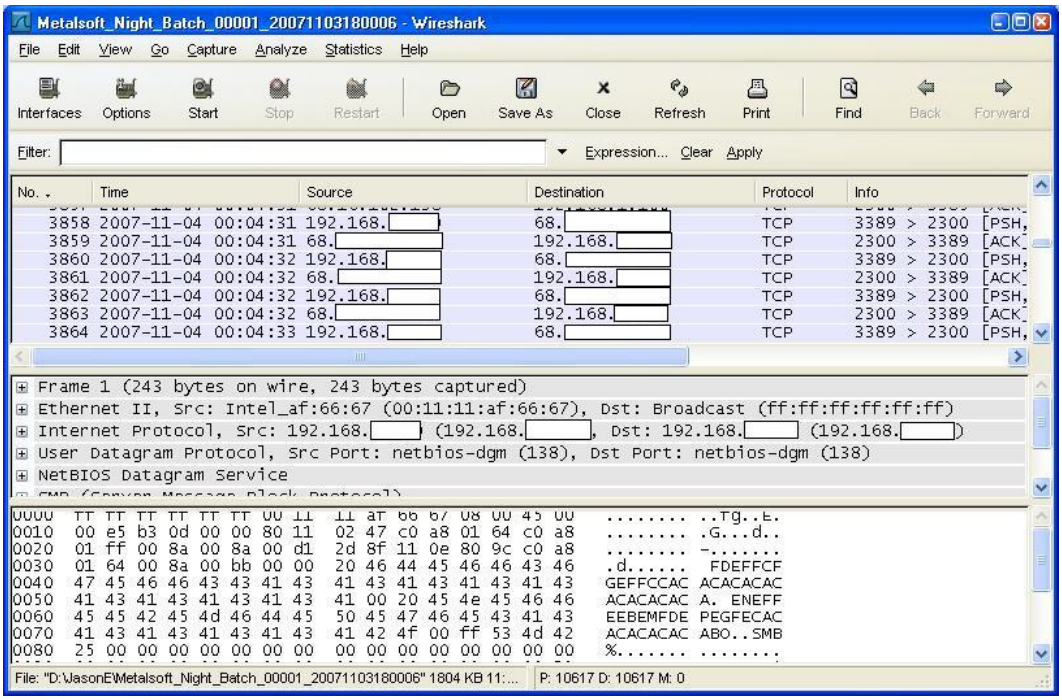


Figure 42: Wireshark main screen

At the top of the screen shot of figure 42 there is filter tool bar. With this toolbar we can filter out data that is not wanted in case we are looking for certain instances. This can display MAC address, protocols, or you can put in expressions to make getting data easier. By clicking on the expression button you will get the window in Figure 43. This example shows we can search for FTP responses that are true, meaning if there was a FTP response.

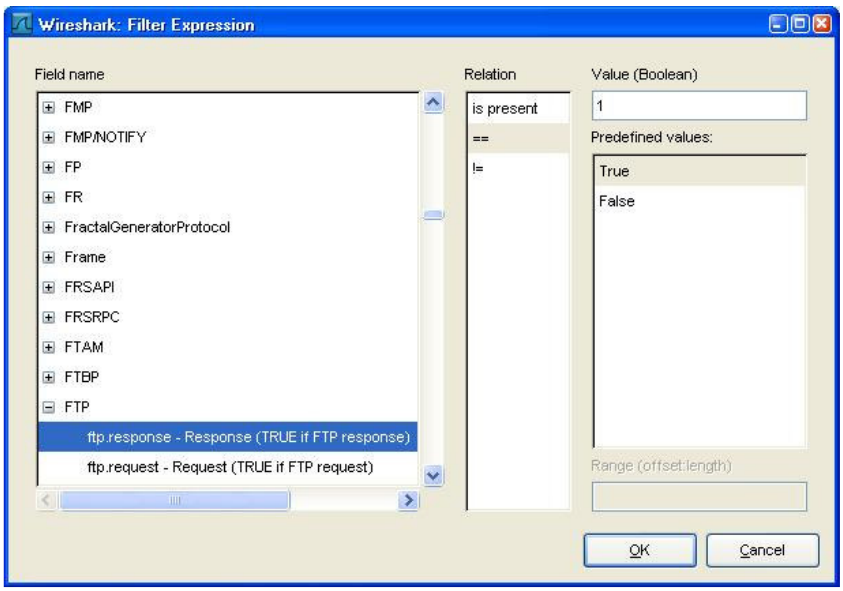


Figure 43: Expression window

The first pane is the packet list pane that displays a summary of each packet captured. This is divided up into six columns. The number column tells what number the packet captured was. The time column shows us the time that packet came in formatted according to whether you choose absolute, relative, or epoch. Then we have the source column telling us where the packet came from. The destination column tells us where the packet was going. The protocol column tells us the protocol whether the packet was TCP, DNS, etc. Finally, there is the info column that tells us the port, SYN or ACK, etc. In this pane the packets can be colorized to make finding certain packets easier. We can customize the coloring to however we please. We can also sort the packets by clicking on the column names. For example, if we want to see all the TCP packets together we can click on protocol and they will all be displayed together.

The pane below the packet list pane is the packet details pane. When we select a packet in the packet list pane the information about the packet is displayed in the packet details pane. The details are displayed in a tree. We can gather a lot of information about the packet. Depending on the packet we can find information such as the size of the packet, MAC addresses of routers the packet has been through, and a lot more detailed information.

The bottom pane is the packet bytes pane. When we select an item in the packet details pain we get hexadecimal and text information about the packet. If the packet was transferred in plain text we can read what was in the packet.

The logs were reviewed for two months to understand what was typical on the Metalsoft network. Typically the traffic was of DHCP requests, IGMP requests, which stands for Internet Group Management Protocol and is a communication protocol used between IP hosts and multicast routers to establish multicast group memberships from the router, and DNS zone transfers. On the average, the log sizes would be between 500 to 1500 kilobytes. After the first two months, the logs that were 1 MB or larger were viewed since this was above the norm. There were several incidences where this happened. Some causes that made the logs large in size were when the server would get its Windows Updates. This seemed to happen on Fridays.

There was one case where someone had logged in through Terminal Services from an IP address that was not familiar. The username or password was not displayed but port 3389 was listed all over the log. This is the port that Terminal Services uses. The IP addres was ran through Sam Spade, which is an integrated network query tool, and traced back to a Holiday INN in Buena Park, CA. This is where Metalsoft's headquarters is located and this is the hotel where Metalsoft employees stay when they have to travel to California. It was discovered that the VP of Sales was in California at that time and he was the one who was logging into the server remotely.

The FTP server was of most concern since we knew this was being attacked. The FTP logs were consulted weekly to see if people were trying to access the server. The FTP logs would give a date and time when someone was logging in. From there we could go to the Wireshark logs and discover what was taking place. This made viewing the logs simple since we had a date and time and would not need to search through the logs to find the occurrence. Wireshark would display information such as what is shown in Figure 44.

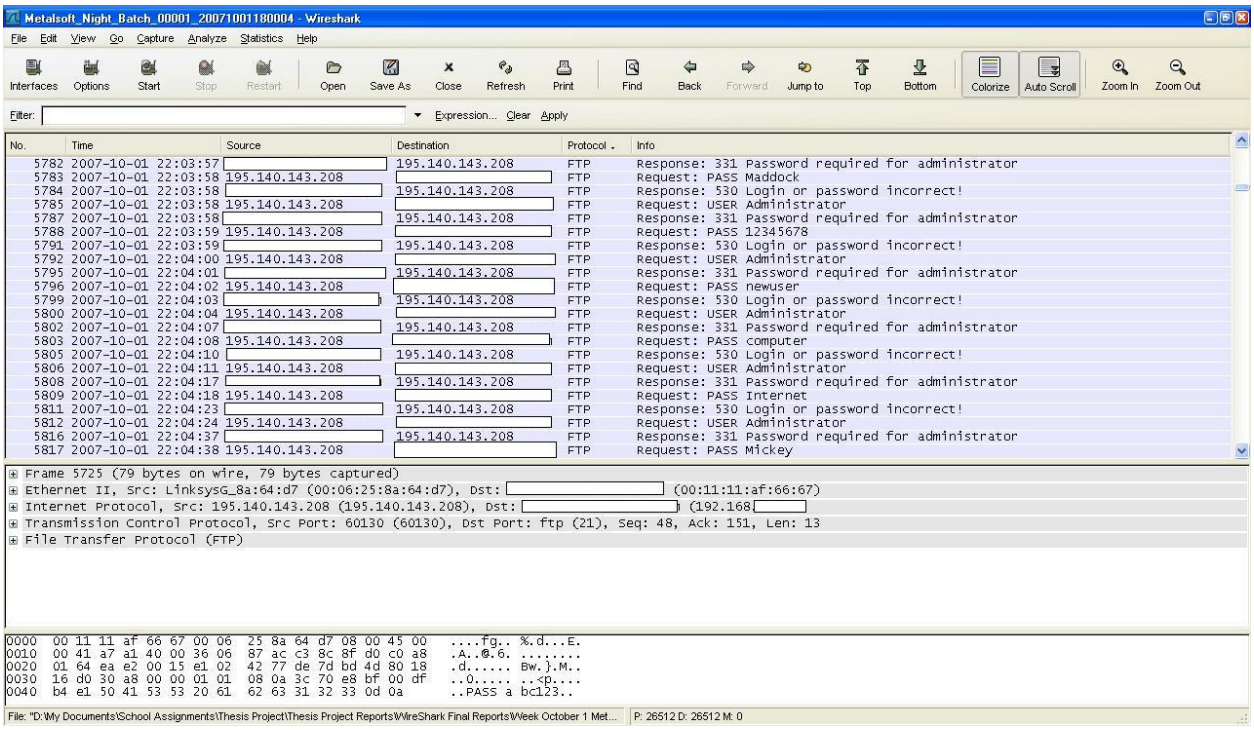


Figure 44: FTP being attacked

From Figure 44 I can see that the FTP was under going a dictionary attack on October 1, 2007 at 11 p.m. CST. Wireshark captured the IP address 195.140.143.208 and this IP is located in Amsterdam from using Sam Spade. There seems to be a lot of attacks from Amsterdam with all different IP addresses. The attacker wants to login as the administrator account. There have been similar attempts like this only using the anonymous login. We are not sure why they are using the passwords they are using but from Wireshark we can see that they are trying passwords as Mickey, Maddock, or newuser. It is understandable why they tried 12345678 or computer since these are simple generic passwords. There have been other attempts to login using usernames such as Sally or Frank. None of these usernames or passwords have anything to do with Metalsoft Inc.

There were discovered attacks from China on the FTP server. In Figure 45 we can observe the Wireshark log of the attack.

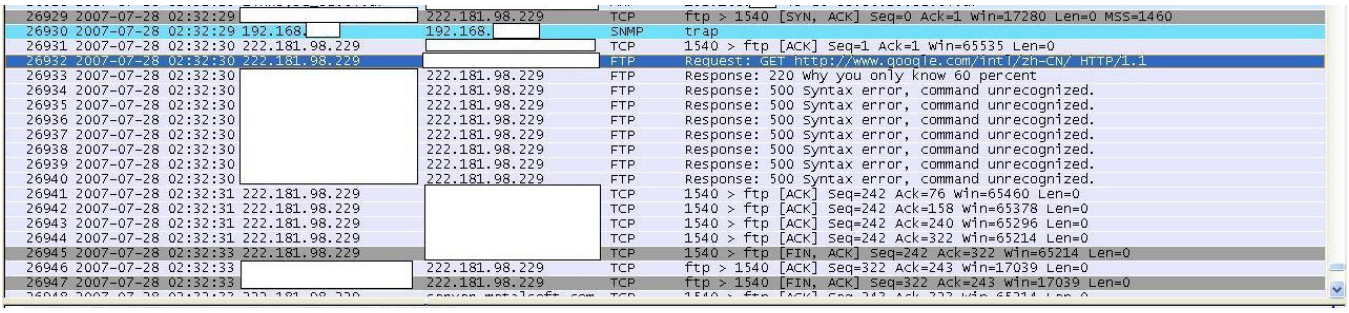


Figure 45: China attack

From the log we can see that someone was trying to log into the FTP on July 28, 2007. The IP address is 222.181.98.229. What was odd about this attempt was they were logging in using the address http://www.google.com/intl/zh-CN/ HTTP/1.1. Research for this is quite difficult. Most of the search pages found with Google show Chinese text. Even after translating the pages, they are still hard to follow. The IP address is in China. From a forum from LinuxQuestions.org, I learned that someone was using an anonymous proxy to get information from the FTP server. (acid_kewpie, 2006, Thread #2) This attack happened quite often to Metalsoft in the summer but has not been seen since August.

For the dictionary attacks there has been a ban on login attempts after five tries to prevent these attackers from entering the FTP. With the China attacks the IP addresses have been banned altogether so if the hackers were to try from that IP address again they would automatically be disconnected.

Review of Wireshark

I think Wireshark is a powerful tool. There are several options available that I did not cover but they did not serve my purpose for this assessment. Wireshark is capable of taking the data and creating graphs. I found the traffic to be steady and for what I was looking for I had other methods of discovering when and where to look. Wireshark offers an expert info option that scans a report and gives warnings and errors but this never came up with anything for me. I was concerned with who was coming in and what they were doing. Wireshark was not as easy to use as Nessus as the options were not all that straight forward. There are some memory issues when loading the log files that would be nice if the developers could fix. However, the information it gives is quite empowering.

Conclusion

Wireshark and Nessus are powerful, free tools for safeguarding a network and identifying how to fortify it. Metalsoft now has some hardware that needs updating because it is no longer supported such as the HP Direct for the network printers. Even though this is a small network and there are not too many problems we did discover some new issues and still have some fixes that need attention. Running the software remotely is a convenience since we all travel and can not be on-site when incidents should happen. There are still a lot of options that need to be explored to better benefit Metalsoft with the software. With Nessus there were high confidence ratings for the server and the salesmen seem to keep up their PCs well these days. The time taken to scan the network is tolerable and the interface is simple to use. Wireshark provided a lot of insight of the network traffic. There are still some issues with the memory usage and compatibility with the GUI color scheme. The server is up and running well and it is apparent that no one is getting into the server that should not be.

I do not know how much further I can take this project as Metalsoft Inc. is merging into Amada America Inc. The Itasca office will be closing and the salesmen will be moving to the Mt. Prospect site on January 1, 2008. When the server comes to Mt. Prospect the server will no longer be public and many of the services that are currently running will no longer be needed. There were plans to incorporate Snort on the server to notify personnel of incidences but this has been terminated. If we are to keep up any of our services on this server Metalsoft will have to ask Amada for permission. Amada provides many of the same services already. There has not been a need for the FTP server in awhile so the FTP server has been shutdown to prevent unwanted users from breaking in. Nessus can not be run from the server on Amada's network since their IDS system would shutdown the scans. All of the system administration will be left to Amada's IT staff.

References

acid_kewpie (2006). *GET http://www.google.com/intl/zh-CN/ ?*. Retrieved November 25, 2007, from LinuxQuestions.org Forum Web site: http://www.linuxquestions.org/questions/linux-security-4/get-httpwww.google.comintlzh-cn-483956/

Archibald, N., Ramirez, G., Rathaus, N., Burke, J., Caswell, B., & Deraison, R. (2005). *Nessus, Snort, & Ethereal Power Tools: Customizing Open Source Security Applications.* Rockland, MA: Syngress Publishing, Inc.

Davis, J. (2007). *KnownBugs/ OutOfMemory*. Retrieved December 2, 2007, from Wireshark.org Web site: http://wiki.wireshark.org/KnownBugs/OutOfMemory

Litchfield, D. (1999). NT *IIS4 Remote Web-Based Administration Vulnerability*. Retrieved November 19, 2008, from Security Focus Web Site: http://www.securityfocus.com/bid/189/discuss

Network Instruments, LLC. (2006). *NIPRSUO1013*. Retrieved November 19, 2007, from Network Instruments, LLC Web site: http://www.networkinstruments.com/support/niprint_eol.html

Tenable Network Security. *The Network Vulnerability Scanner*. Retrieved October 29, 2007, from Tenable Nessus Web Site: http://www.nessus.org/nessus/