Distributed Vulnerability Assessment with Nessus

Faiz Ahmad Shuja

GIAC Security Essentials Certification (GSEC) Practical Assignment
Version 1.4b (Option 2)
October 2003

Table of Contents

Abstract

This paper is based on a scenario which I implemented in my network. It discusses the step-by-step implementation of a distributed vulnerability assessment setup using Nessus. Vulnerability Assessment (VA) of my organization's network comprises one of the major tasks I have to perform on a weekly basis. In the old centralized setup, the efficiency of the assessment decreased as the size of the network increased. For a more complete and reliable assessment, it was necessary to change the way we ran the tests, because of the size of the network, and the many different zones that have to be scanned and examined, the centralized VA setup was inappropriate. It was therefore decided that a distributed VA setup be implemented using Nessus with proper attention given to the placement of the daemons, network architecture and user management. This paper attempts to describe the way in which this transition was accomplished.

1. Introduction

Since networks are growing at a fast pace, it is hard to assess large networks from a single location. In large networks, the task of vulnerability assessment (VA) can be distributed by deploying VA agents in different zones of the network and controlling them from a single central location. This makes the VA tasks faster and easier to manage and control. This type of vulnerability assessment is known as Distributed Vulnerability Assessment.

> The best way to protect yourself against the threat vectors is to regularly audit your *perimeter security* (Cole, Fossen, Northcutt, Pomeranz, p.724).

Nessus is the best vulnerability scanner that fits to our needs of faster and manageable vulnerability assessment. Nessus is one of the most powerful, up-to-date and free vulnerability scanners around. It has a client / server architecture, which allows us to deploy a distributed vulnerability assessment scenario. It does not rely on standard port descriptions like other scanners, rather it assesses the application running on that port. Another reason behind selecting Nessus is that it uses a plug-in based architecture. Each security test is an external plug-in. Nessus can be updated with latest plug-ins when required or we can write our own tests using NASL (Nessus Attack Scripting Language) as an external plug-in. Also each plug-in links to CVE for getting detailed information about the security test. Nessus can produce reports in different formats, which can be compared with each other. So Nessus fulfils most of the recommendations made by the SANS Security Essentials course book for a vulnerability scanner.

2. Before

2.1 Centralized / Non-distributed VA

I had started vulnerability assessment of the network when I joined the company a couple of years back. The size of the old network was small as compared to the current one, and the vulnerability assessment setup fulfilled our requirements adequately. As the network size and vulnerabilities started growing rapidly, the task of vulnerability assessment became slow and time consuming. The reason of course, was the non-distributed approach we were using which was designed for small network s.

Due to the lack of knowledge in securit y assessment, I used to be scanning and assessing the servers in network individually. As the size of old network was small, we had ample time to secure the servers all day. The vulnerability assessment task was done dedicatedly which made it time consumin g. The manual scanning of multiple hosts or a subnet is a tough task. The advantage of the old approach was that the scans were thorough and detailed. But due to unavailability of a network assessment policy , I had to wait for the manager's approval for sc anning. Once assessment was completed , I had to distribute the report to the related people.

As the network size started growing rapidly, I started overlooking some of the subnets and was not able to assess the network with same consistency as before.

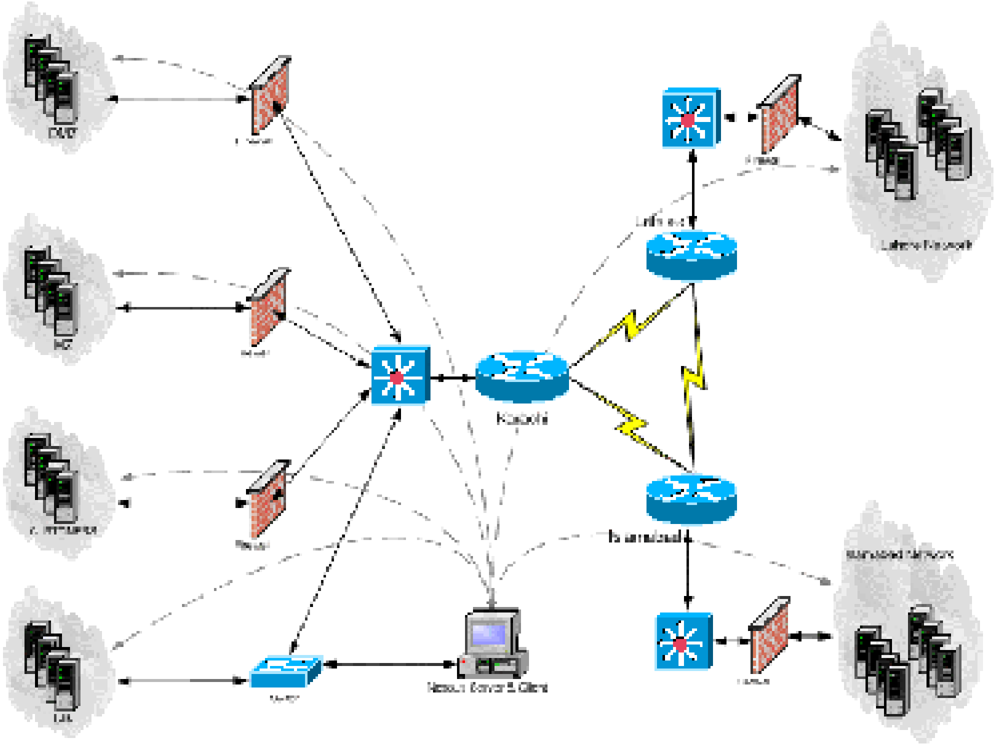


Figure 1 - Old Scenario – Non-distributed Vulnerability Assessment

Figure 1 shows the old network assessment scenario . I had to scan the following zones and networks:

1. Karachi Network
    a. MZ (Militarized Zone) – Critical Servers
    b. DMZ (De-Militarized Zone) – External Servers
    c. Customers Zone – Customer Collocated Servers
    d. LAN - Our Local Network
2. Islamabad Network
3. Lahore Network

As you can see in Figure 1, I had placed the VA machine in our LAN. All the tools required to run the scans were installed on it. The LAN is an external network (like internet) to the different zones. So we normally did not get any internal threats. Whenever we wanted to do an internal assessment we had to place the VA machine inside the specific zone because allowing everything from the VA machine on the firewall was not a good idea. This was dirty task. In effect, we were just protecting the network from outside attacks. We had no assessment measures for an insider attack from the local network or from local systems.

As the network was expanded country-wide, I had to scan the networks in different cities also. Those scans were slow and time consuming because of the multiple hops required. Also pumping high amount of traffic on the network is not a good idea, since it may cause network to choke.

3. During – Implementation

3.1 Design

Because of the server constraints in the old scenario, I planned to change it to distributed vulnerability assessment scenario.

I planned to deploy vulnerability scanning servers on different zones and manage them centrally. The best solution that fit to our needs was Nessus . It was decided to place Nessus servers on the different zones to be analyzed to get the internal view of that network and the systems in it, hence facilitating comparison with an external view.

3.2 Architecture

The implementation for this distributed vulnerability assessment setup is also a distributed architecture. The distribution is as follows:

Server – Nessus Daemon
Web Interface – InProtect Nessus & Nmap Web Interface and MySQL server
Client – Web Browser

Figure 2 shows our setup distributed over three spots: the server, the web interface and the client. This makes a standard client server model setup.

3.2.1 Server – Nessus Daemon

This layer has multiple Nessus daemons placed in the different zones to be assessed. The Nessusd (Nessus Daemon) is the core part of this setup as it performs the vulnerability assessments. The scans are controlled and configured from the web interface. The scan information and plug-ins are stored on the server. The reports are saved in the database on the web interface machine once the scans are completed. Nessusd listens on port 1241 for connections. The communication between the server and web interface is encrypted and authenticated.

3.2.2 Web Interface

This layer has the InProtect Web Interface for Nessus and Nmap running on Apache web server. It is also running MySQL database server for storing assessment reports. This machine acts as a client for the Nessusd machine. It is used to configure, control, and monitor scans. It has the capability to save all scan configurations and vulnerability assessment reports in the database.

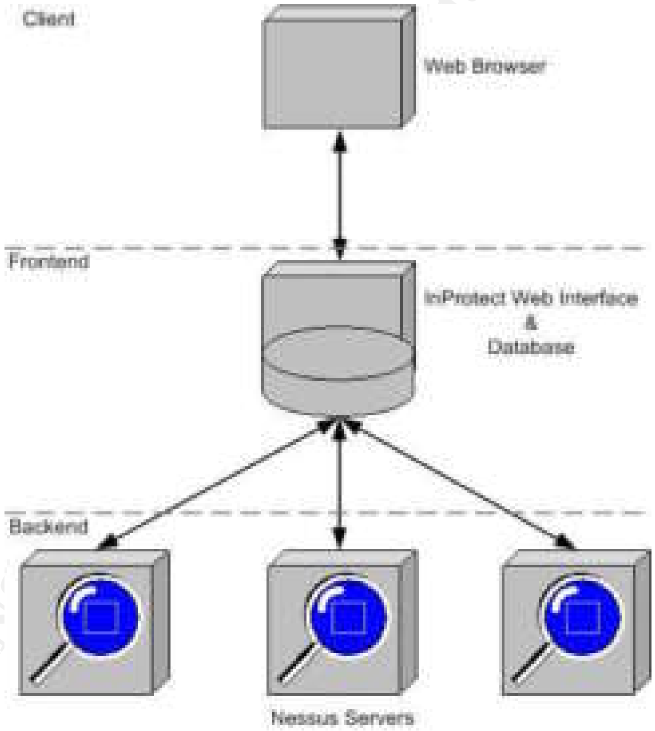


Figure 2 - Architecture

InProtect Web Interface enables you to do scan configurations, plug -ins configuration, scan scheduling, user management, and report management.

The web interface is detached from the server once it has configured and sent the scan request. Once the scan is completed , the results are saved in the database and can be viewed anytime to perform an analysis. The reports can be viewed in HTML or PDF format using the web browser.

3.2.3 Client

The client for this setup is a web browser. Authorized users are allowed to logon to the InProtect Web Interface. Once the user is logged in, scan can be configured and started. Once an assessment is completed, reports can be viewed and analyzed.

4. Deployment

4.1 Required software

The packages needed to make this setup work on RedHat Linux 8.0 are: The vulnerability scanner Nessus, port scanner Nmap, Apach e web server with PHP, MySQL database server with PHP support, Perl and InProtect web interface.

- Red Hat Linux 8.0
  ftp://ftp.redhat.com

  Most of the packages like webserver, PHP, database server, and Perl come with Red Hat Linux. They can be installed separately or can be updated after installation. So I installed Apache, MySQL, PHP and Perl while installing Red Hat. After that I updated the packages by apt.

- Apache Web Server - for management and report distribution
  http://www.apache.org

- PHP – required by the web interface
  http://www.php.net

- MySQL – for saving reports and plug -ins information
  http://www.mysql.com

- Perl – required by InProtect scripts
  http:// www.perl.com

- Nessus – the vulnerability scanner
  http://www.nessus.org

- Nmap – the port scanner used by Nessus
  http://www.insecure.org/nmap

- InProtect – the web interface for controlling Nessus and Nmap
  http://www.inprotect.com

4.2 Backend Installation

Following is the installation I di d for the Nessus servers that are placed in different zones, as you can see them in the Figure 3.

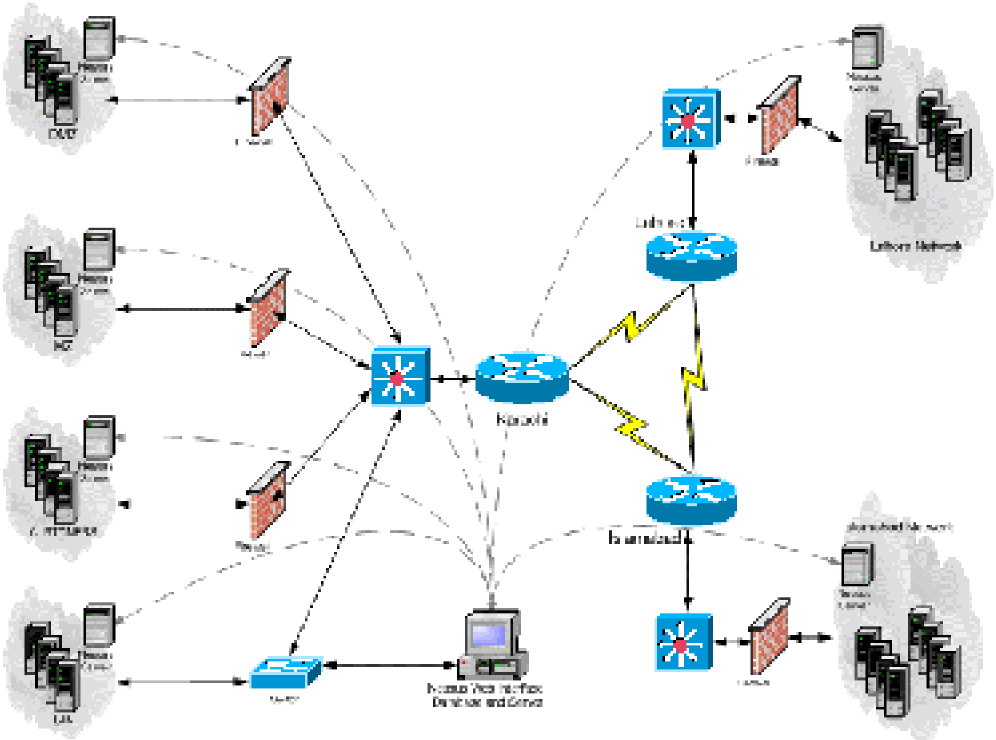


Figure 3 - New Sce nario – Distributed Vulnerability Assessment

4.2.1 RedHat 8.0 – Installation

The following options were selected once I got the Welcom e Screen.

1. Your Language > Next
2. Your Keyboard configuration > Next
3. Your Mouse configuration > Next
4. Installation Type > Custom > Next
5. Disk Portioning Setup > Automatically partition > Next
    a. Automatic Portioning > Remove all partitions on this system > Next
    b. Disk Setup > Next
6. Boot Loader Configuration > Next
7. Advanced Boot Load Configuration > Next
8. Network Configuration > Enter your network details > Next
9. Firewall Configuration > No Firewall > Next
10. Language Support Selection > Next
11. Time Zone Configuration > S et your time zone > Next
12. Account Configuration > Enter root password > Next
13. Authentication Configuration > Next
14. Package Group Selection

   a. Editors > Select all
   b. Text-based Internet > Lynx
   c. System Tools > Nmap
   d. Check Select individual packages > Next
   e. System Environment > Libraries > Select all in group
   f. Next

15. Preparing to install > Next
16. Boot Disk Configuration > Skip > Next
17. Graphical Interface (X) Configuration > Select your card > Next
18. Monitor Configuration > Select your monitor > Next
19. Installation Complete > Next

After logging into the machine, I did some OS hardening , disabled the unused services and configured IPTables.

4.2.2 Nessus – Installation

Following are the steps taken for Nessus installation .

```
# mkdir nessus
# cd nessus/
```

Downloaded Nessus through lynx web browser

```
# lynx http://ftp.nessus.org/nessus/ nessus -2.0.7/src/nessus -
core -2.0.7.tar.gz
# lynx http://ftp.nessus.org/nessus/nessus -2.0.7/src/nessus -
libraries -2.0.7.tar.gz
# lynx http://ftp.nessus.org/nessus/nessus -2.0.7/src/libnasl -
2.0.7.tar.gz
# lynx http://ftp.nessus.org/nessus/nessus -2.0.7/src/nessus -
plugins -2.0.7.tar.gz
```

Decompressed and installed the downloaded Nessus files.

```
# tar zxvf nessus -libraries -2.0.7.tar.gz
# cd nessus -libraries
# ./configure
# make
# make install

# tar zxvf libnasl -2.0.7.tar.gz
# cd libnasl
# ./configure
# make
# make install

# tar zxvf nessus -core -2.0.7.tar.gz
# cd nessus -core
# ./configure
```

Got the following error while installing the Nessus

```
configure: warning: Only gtk+ -2.0 was found : the client will
be built but will be extreme ly buggy. Install gtk+ -1.2 if you
```

```
      want stability. If  you do not understand why you would want to
      install version 1.2.x when you have 2.0.x  send a mail to the
      GTK+ team (www.gtk.org) and complain about their inability to
      handle backward compatibility
```

Since GUI client was not needed on the server, I installed it with following options

```
      # ./configure  –disable -gtk
      # make
      # make install

      # tar zxvf nessus -plugins -2.0.7.tar.gz
      # ./configure
      # make
      # make install
```

4.2.2.1 Adding User for Nessus

This is used by the web interface to connect to Nessus server.

```
      # nessus -adduser

Using /var/tmp as a temporary file holder

Add a new nessusd user
----------------------

Login : nessus
Authentication (pass/cert) [pass] :
Login password : *******

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

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

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


Login             : nessus
Password        : *******
DN                 :
Rules             :


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

4.2.2.2 Creating Nessus SSL Certificate

```
      # nessus -mkcert

/usr/local/var/nessus/CA created
```

```
/usr/local/com/nessus/CA created
---------------------------------------------------------------------
                  Creation of the Nessus SSL Certificate
----------------------------------------  -----------------------------


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

CA certificate life time in days [1460]:
Server certificate life time in days [365]:
Your country (two letter code) [FR]:
Your state or province name [none]:
Your location (e.g. town) [Paris]:
Your organization [Nessus Users United]:
---------------------------------------------------------------------
Creation of the Nessus SSL Certificate
---------------------------------------------------------------------

Congratula tions. Your server certificate was properly created.

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

The following files were created :

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

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

Press [ENTER] to exit
```

4.2.2.3 Updating Plug -ins

Then, I updated the plug -ins from the Nessus website.

```
# nessus -update -plugins
```

4.2.2.4 Installing Nmap

Nmap is used by Nessus for port scanning. Downloaded and installed the scanner.

```
# lynx http://download.insecure.org/nmap/dist/nmap  -3.48.tgz
# tar zxvf nmap
# ./configure
# make
# make install
```

4.2.2.5 Starting Nessus Daemon

All done, started the Nessus Daemon.

```
# nessusd -D
```

4.3 Front-end Installation

The following is the installation for Web Interface and Database server. I also installed Nessus on this machine for the external assessment. Since the machine was powerful , it was able to handle everything properly.

4.3.1 RedHat 8.0 – Installation

Following options were selected once I got the Welcome Screen.

1. Your Language > Next
2. Your Keyboard configuration > Next
3. Your Mouse configuration > Next
4. Installation Type > Custom > Next
5. Disk Portioning Setup > Automatically partition > Next
    a. Automatic Portioning > Remove all partitions on this system > Next
    b. Disk Setup > Next
6. Boot Loader Configura tion > Next
7. Advanced Boot Load Configuration > Next
8. Network Configuration > Enter your network details > Next
9. Firewall Configuration > No Firewall > Next
10. Language Support Selection > Next
11. Time Zone Configuration > Set your time zone > Next
12. Account Confi guration > Enter root password > Next
13. Authentication Configuration > Next
14. Package Group Selection
    a. Editors > Select all
    b. Text-based Internet > Lynx
    c. Servers > Server Configuration Tools > Select all
    d. Servers > Web Server > Select all
    e. Servers > SQL Database S erver > Select all
    f. System Tools > Nmap
    g. Check Select individual packages > Next
    h. System Environment > Libraries > Select all in group
    i. Next
15. Preparing to install > Next
16. Boot Disk Configuration > Skip > Next
17. Graphical Interface (X) Configuration > Select you r card > Next
18. Monitor Configuration > Select your monitor > Next
19. Installation Complete > Next

4.3.2 Upgrading – Packages

I used apt for upgrading the packages used in this se tup.

4.3.2.1 Installing Apt

```
# mkdir /dva
# cd /dva/
# mkdir apt
# cd apt
# lynx
http://ftp.freshrpms.net/pub/freshrpms/redhat/8.0/apt/apt-
0.5.5cnc6-fr0.rh80.1.i386.rpm
# rpm -ivh apt-0.5.5cnc6-fr0.rh80.1.i386.rpm
# apt-get update
```

4.3.2.2 Upgrading Web Server

```
# apt-get install httpd
```

4.3.2.3 Upgrading PHP and MySQL

```
# apt-get install php-mysql
# apt-get install mysql
```

4.3.2.4 Upgrading Perl

```
# apt-get install perl
```

4.3.3 Installing InProtect Web Interface

InProtect requires following packages to run:

1. Web Server
2. PHP
3. MySQL
4. Perl
5. Nessus
6. Nmap

I already had installed and updated web server, PHP, MySQL and Perl.

```
# cd /dva
# mkdir inprotect
# cd inprotect
# lynx
http://twtelecom.dl.sourceforge.net/sourceforge/inprotect/inpro
tect_014.tar.gz
# tar zxvf inprotect_014.tar.gz
```

4.3.3.1 Configuring MySQL for InProtect Database

```
# /etc/rc.d/init.d/mysqld start
# mysql -u root
```

Setting password for user 'root'

```
> set password for 'root'@'localhost'=password('password');
```

Creating InProtect MySQL database

```
# cd sql
# mysql  -h localhost  -u root  -p < inprotect.sql
```

4.3.3.2 Configuring Perl modules required for InProtect

```
# perl  -MCPAN -e 'install DBI'
# perl  -MCPAN -e 'install MIME::Lite’
# perl  -MCPAN -e 'install Parallel::ForkManager'
# perl  -MCPAN -e 'install Date::Calc’
```

4.3.3.3 Setting up InProtect PHP pages

```
# cd..
# cd html
# mkdir /var/www/html/nessus
# cp –R * /var/www/html/ nessus/
```

Configuring config.php

```
# cd /var/www/html/nessus/
# pico config.php

  $dbtype="mysql";
  $dbhost="127.0.0.1";
  $dbuname="root";
  $dbpass="password";
  $dbname="inprotect";
```

4.3.3.4 Setting up InProtect scripts

```
# cd /dva/inprotect
# cd scripts
# cp * /usr/local/bin
```

Configuring inprotect.cfg

```
# cd /usr/local/bin
# pico inprotect.cfg

          #Nessus user you created in Nessus
          NESSUSUSER=nessus

          #Nessus password
          NESSUSPASSWORD=password

          #Nessus server hostname or IP address
          NESSUSHOST=127.0.0.1

          #Database name
          DATABASENAME=inprotect

          #Database host
          DATABASEHOST=localhost
```

```
                    #Database user
                    DATABASEUSER=root

                    #Database password
                    DATABASEPASSWORD=password

                    #Scan results URL
                    RESULTURL=http://localhost/nessus/


        # chmod 640 inprotect.cfg
```

4.3.3.5 Creating directories

Following are directories needed by InProtect.

```
        # mkdir /usr/data
        # mkdir /usr/data/nessus
        # mkdir /usr/data/nmap
```

4.3.3.6 Creating cron jobs for InProtect scripts

```
        # crontab  -l > /tmp/cron.tmp
        # cat crontab >> /tmp/cron.tmp
        # crontab /tm p/cron.tmp
        # rm /tmp/cron.tmp  -rf
```

4.3.3.7 Popu lating InProtect database with Nessus plug -ins

```
        # updateplugins.pl
        # tail -f /var/log/updateplugins
```

4.3.3.8 Customizing the Web Interface

Currently InProtect Web Interface does not provide an op tion to select the Nessus Server. It can only be configured in inprotect.cfg present in /usr/loca/bi n. Since I had to manage multiple servers from a single location , I did not want to change the ser ver IP from shell each ti me before scanning. So I made a web page in PH P using InProtect 's existing include files to configure the inprotect.cfg file. It enables you to configure the server IP, username, and pass word and writes the changes to inprotect.cfg. Once the scan starts, the file can be changed again for the new scan as the web interface is detached from the server. I also made some changes in header.php and footer.php according to my organization 's requirements, wh ile making sure that InProtect is proper ly credited.

4.3.3.9 Accessing the Web Interface

I logged on to http://web-interface -ip/nessus/ through a web browser.

Login: admin
Pass: password

Once logged in, change d the password from

Settings > My details

4.3.4 Scanning Configuration

4.3.4.1 Profiles

One of the reasons behind selecting InProtect for this scenario was its ability to create multiple scan configurations (profiles) and schedul e them.

I created the following profiles from Settings > Nessus Scan Profiles > Create New Profile :

- MZ – scans the serve rs inside MZ
- DMZ – scans the servers inside DMZ
- Corporate – scans the servers inside Corporate network
- LAN – scans the clients inside LAN
- Islamabad – scans the servers and c lients in Islamabad network
- Lahore – scans the servers and clients in Lahore networ k
- Network – scans the network appliances (routers, switches, access servers, etc)
- Firewall – port scans the servers behind firewalls to verify the fire wall policies.

I did not enable DoS based plug-ins in the above profi les to avoid any interruption in the production network. I separately created a profile with all plug-ins enabled to assess the perimeter security externally i.e. known as penetration testing (Pen-Test).

4.3.4.2 Users

There are four levels of users in InProtect. All of them can run the sc ans as well as view the reports.

- Level 1 – can manage profiles and users (admin)
- Level 2 – can manage profiles only
- Level 3 – can manage users only
- Level 4 – can run scans and view reports only

I had created the scanning profiles with Level 1 user a nd assigned it to all profiles earlier. Then I created separate Level 4 users for each profile and assigned them to their corresponding profiles. Also created separate Level 4 users for each client in the Corporate profile so that every client can have its own separate user.

The reason for creating separate users for each profile was to easily manage the report distribution. If I had created single user for all profiles then everyone would be able to see reports of all profiles, which was undesirable .

4.3.4.3 Scanning & Scheduling

After I was done wi th profile and user creation, I logged in with the users I had created earlier and configured the scanning and scheduling policies. I added the hosts list according to user's zone and profile .

I did not set any scheduling on any Nessus server to avoid interruption to services because of unmonitored scanning. I can enable 'Safe Checks' option in Nessus to avoid interruption of services but I prefer to do monitored vulnerability assessment weekly. So, I start t he assessment by selecting 'Run now' in 'Manage Schedule' interface after I have selected the Nessus server to scan from. I scheduled to do an external port scan daily in off -peak hours of the servers and clients inside different zones to make sure their policies are working and there are no unwanted ports open on them.

4.3.4.4 Port Scan

I can also perform port scan s from Security Scan> Nmap Port Scan

4.4 Reports

When Nessus completes the assessment , it saves the reports in the database which can be vi ewed from the 'Reports' menu. InProtect enab les us to fetch the report in HTML and PDF format s. The reports are produced with detailed information showing the risk level of vulnerabilities found on the services running. It also tells you the steps to be ta ken for preventing them from being exploited. Also combined reports of that profile can be searched on the basis of severity, host name, service, and nessus plug -in. InProtect shows you scan trend graphs by date and scan.

5. After

5.1 Enhancements

The o verall state of security was enhanced after I deployed the distributed vulnerability assessment scenario in my organization. Some of the advantages and enhancements are as follows:

5.1.1 Policy

The SANS Security Essentials course book teaches us to have a policy for every information security procedure. Earlier there was no security policy for vulnerability assessment and other security procedures. If anything would have gone wrong, I would have been held responsible for it. So I formulated a security pol icy and agree ment for the proced ures I used to perform. I created a detailed policy for vulnerability assessment which included policy for zones, schedule, level of assessment and other related details.

5.1.2 Automation & Scheduling

The feature of scheduling enabled us to automate the task of vulnerability assessment. I can schedule the assessment on daily, weekly and monthly basis according the system's criticality. We have more consistency by scheduled scanning. In a manual scan, you normally ignore some systems. It helps me in discovering new un-patched machines which are brought up on the network without permissions.

5.1.3 Internal / External Assessment

The new setup has the ability to give internal and external view of the network. It scans systems inside different zones to get a detailed internal view as compared to an external one. We can then compare it with an external port scan or assessment.

5.1.4 Report Management & Aggregation

Report management is easy in this scenario as all the reports are stored at central location and can be shared accordingly. The web interface enables us to aggregate and search multiple reports which help in finding the vulnerability severity, trends, and differences.

5.1.5 User Management

The web interface gives us the ability to manage multiple users and distribute the profiles accordingly. Now I do not have to print or email the report to related people. They can just login to web interface and view the reports.

5.1.6 Awareness

The advantage of this setup was that I started pointing out the problems regularly to the related authorities, especially on LAN. The result of this practice was that people started contacting me for security related issues and avoided running unauthorized applications on their systems .

5.2 Complications

There are some complications and issues in this scenario as they exist in a distributed environment. Some of them are as follows:

5.2.1 Patch Management

I have to manually update all the servers for latest patches and upgrades which is time consuming.

5.2.2 Plug-ins Management

Though I have configured the Nessus plug-ins to be updated in database automatically through cron but I have to verify it on all servers.

5.2.3 Users Rights

There are still certain issues with the current u ser management . All users can run and view the scans. There should also be a view only user. If someone would scan through his profile, he would be violating the scanning policy though, as I a m the only one who is allow ed to run a net work scan.

5.2.4 Sche duling

Currently it does not have the ability to schedule a scan on multiple Nessus servers at the same time. I have to manually start the scan, though I p refer that also.

5.3 Future Plans

I plan to enhance the overall state by setting up more secure us er management and scheduling features.

Conclusion

It is observed now that the VA is faster, reliable and far more efficient than before. We can scan the internal as well as external networks without much hassle. All the reports and configurations are sto red in a central location thus facilitating easy retrieval and the user management feature of web interface enables us to customize the data access. Even though a distributed setup is difficult to maintain and update, the pros outweigh these cons.

Appendix

Screenshots

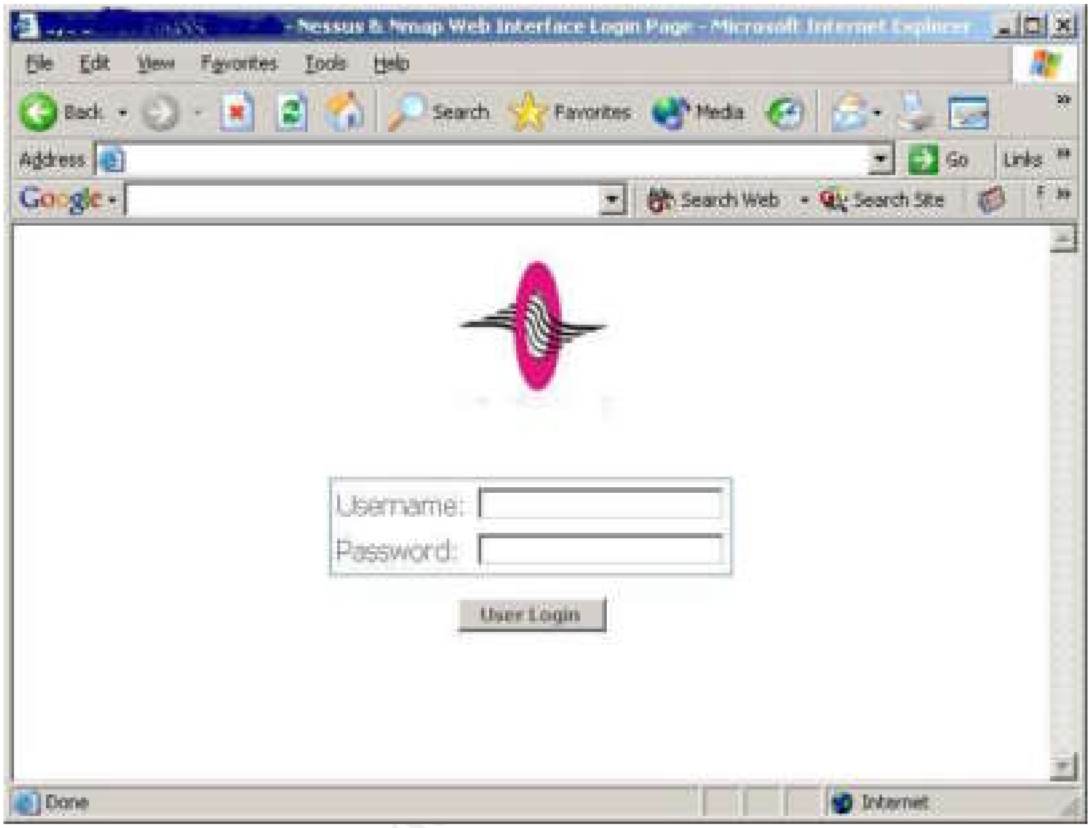
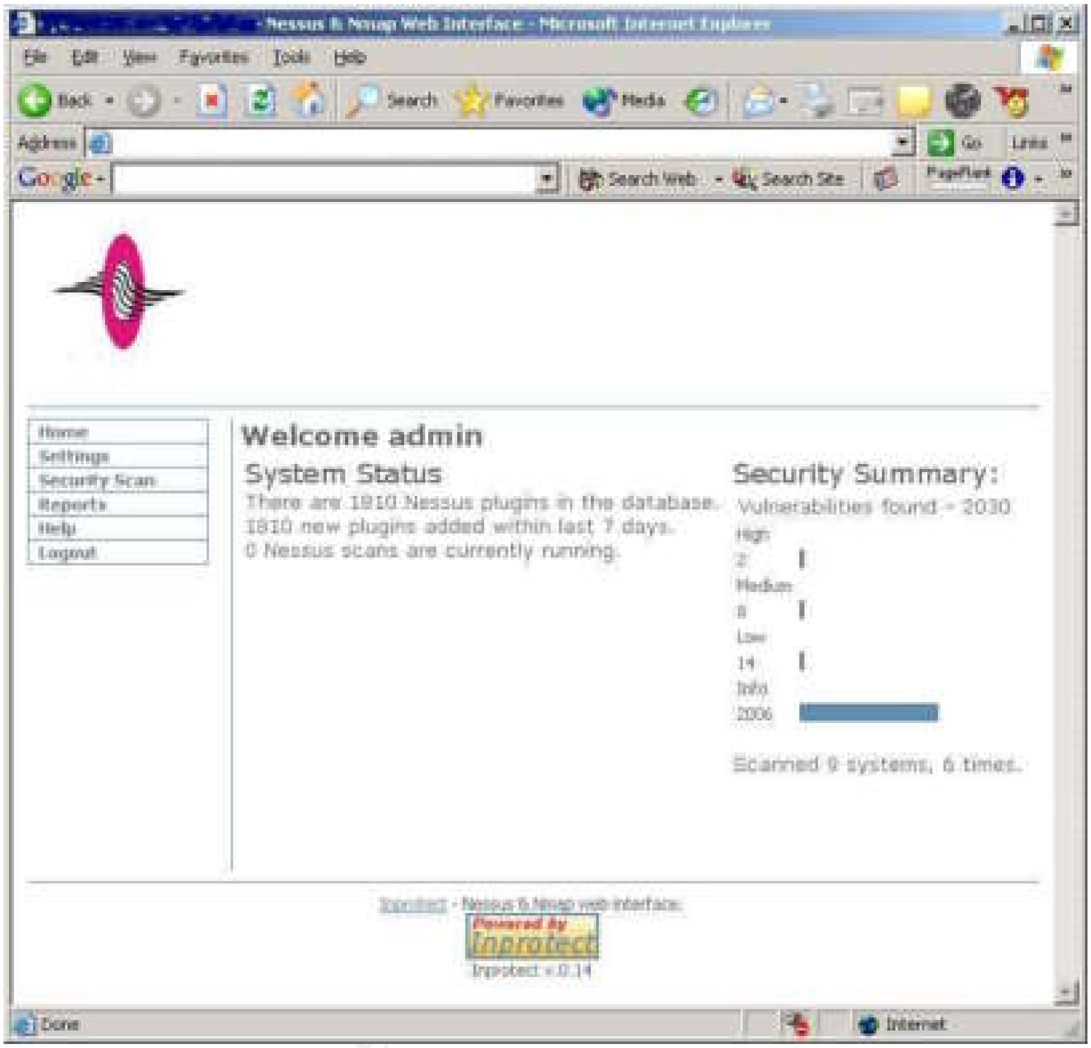

Nessus & Nmap Web Interface Login Page - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address Go Links
Google Search Web Search Site
Username:
Password:
User Login
Done
Internet

Figure 4 – Login Page


Nessus & Nmap Web Interface - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address Go Links
Google Search Web Search Site PageRank
Home
Settings
Security Scan
Reports
Help
Logout
Done
Internet

Welcome admin

System Status

There are 1810 Nessus plugins in the database.
1810 new plugins added within last 7 days.
0 Nessus scans are currently running.

Security Summary:

Vulnerabilities found - 2030

High
2
Medium
8
Low
14
Info
2006

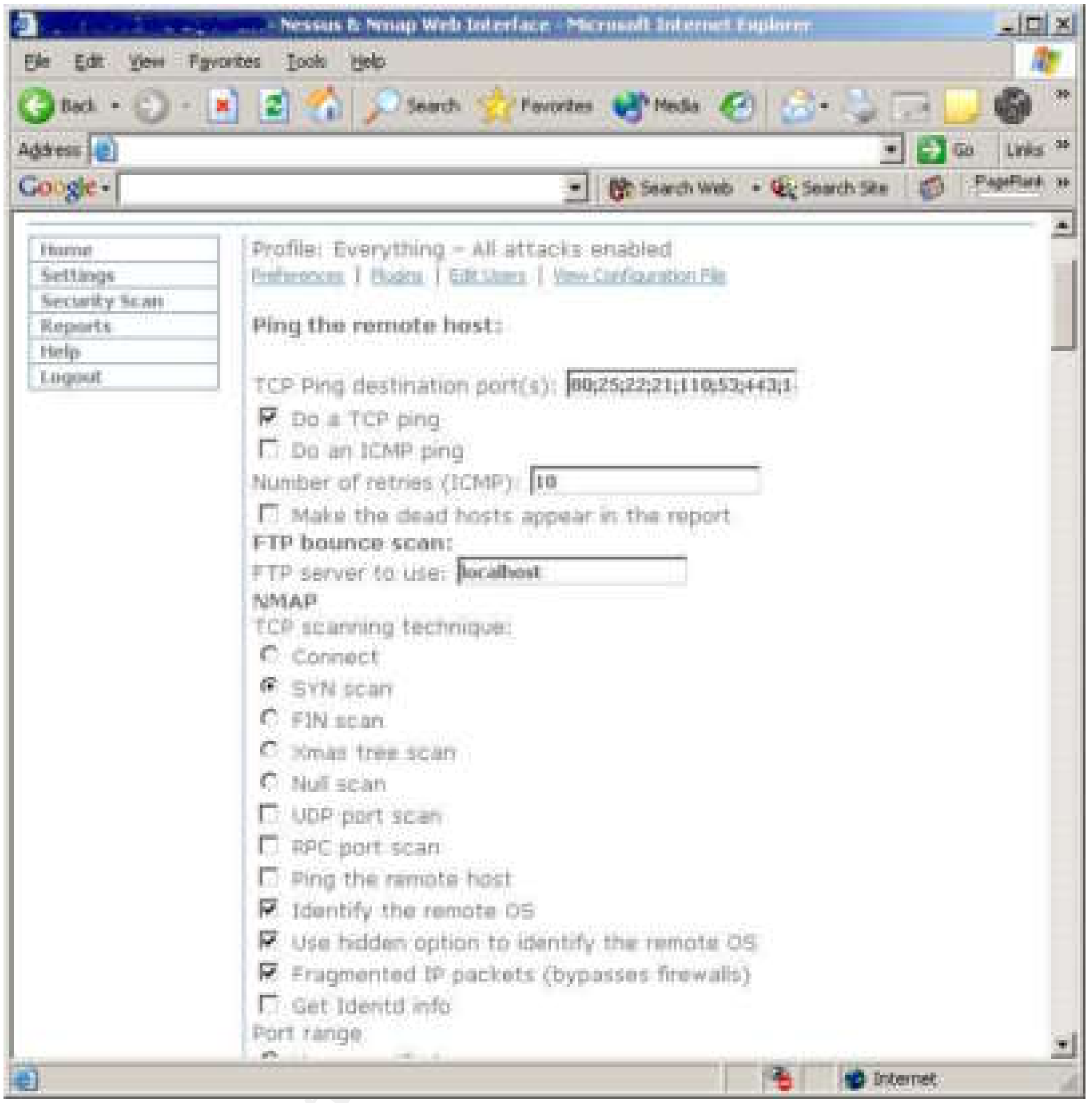
Scanned 9 systems, 6 times.

Inprotect - Nessus & Nmap web interface.

Powered by Inprotect

Inprotect v.0.14

Figure 5 – Welcome Scr een


Nessus & Nmap Web Interface - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address Go Links
Google Search Web Search Site PageRank
Home
Settings
Security Scan
Reports
Help
Logout
Internet

Profile: Everything - All attacks enabled

Preferences | Plugins | Edit Users | View Configuration File

Ping the remote host:

TCP Ping destination port(s): 80;25;22;21;110;53;443;1

☑ Do a TCP ping

□ Do an ICMP ping

Number of retries (ICMP): 10

□ Make the dead hosts appear in the report

FTP bounce scan:

FTP server to use: localhost

NMAP

TCP scanning technique:

○ Connect

● SYN scan

○ FIN scan

○ Xmas tree scan

○ Null scan

□ UDP port scan

□ RPC port scan

□ Ping the remote host

☑ Identify the remote OS

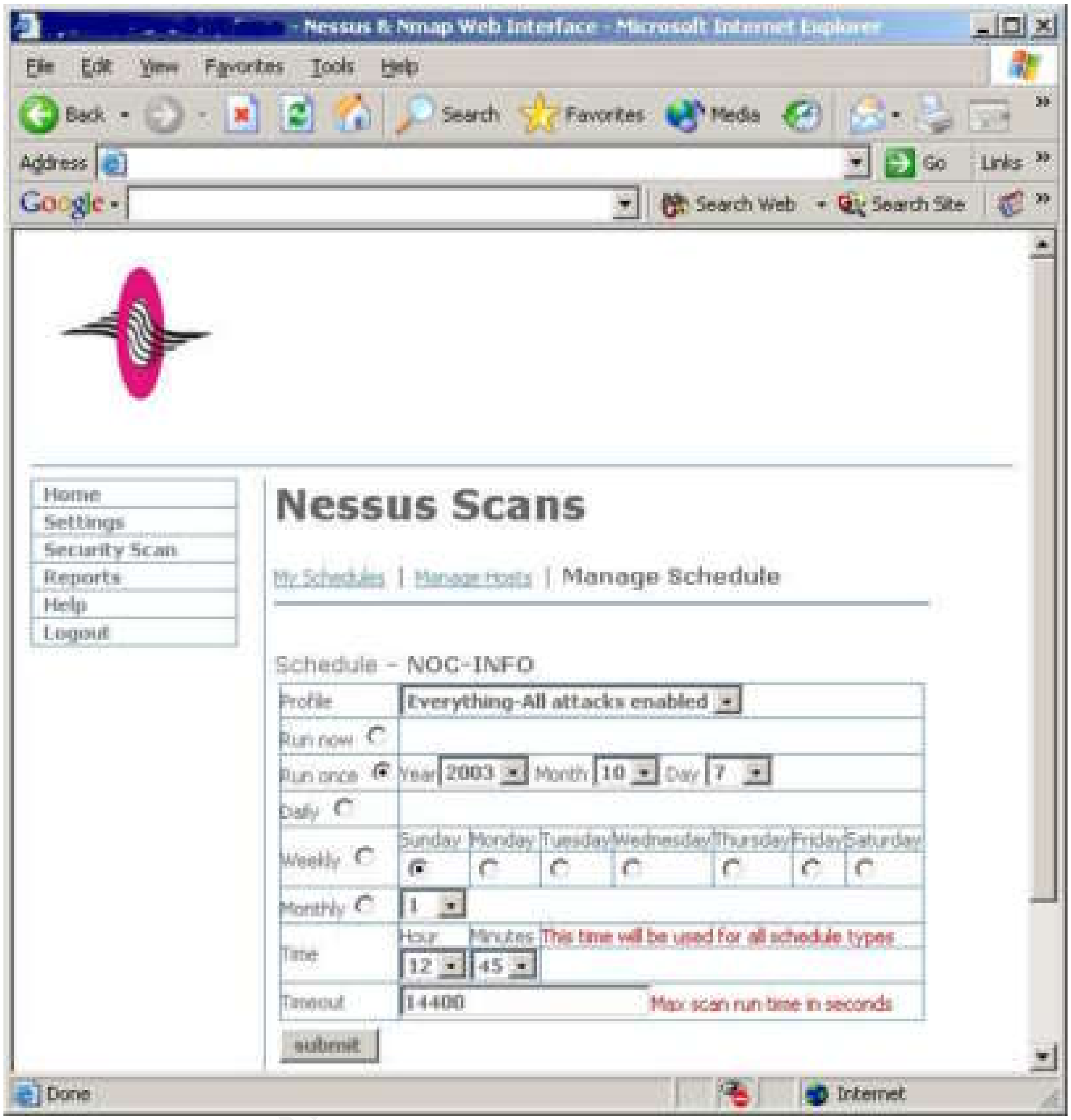
☑ Use hidden option to identify the remote OS

☑ Fragmented IP packets (bypasses firewalls)

□ Get Identd info

Port range

Figure 6 – Profile Configuration


Nessus & Nmap Web Interface - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address Go Links
Google Search Web Search Site
Home
Settings
Security Scan
Reports
Help
Logout
Done
Internet

Nessus Scans

My Schedules | Manage Hosts | Manage Schedule

Schedule - NOC-INFO

Profile	Everything-All attacks enabled
Run now ○	
Run once ●	Year 2003 Month 10 Day 7
Daily ○	
Weekly ○	Sunday ● Monday ○ Tuesday ○ Wednesday ○ Thursday ○ Friday ○ Saturday ○
Monthly ○	1
Time	Hour 12 Minutes 45 This time will be used for all schedule types
Timeout	14400 Max scan run time in seconds

submit

Figure 7 – Scan Configuration

References

1. Cole, Eric. Fossen, Jason. Northcutt, Stephen. Pomeranz, Hal . SANS Security Essentials with CISSP CBK, Volume One. Reading: SANS Press, April 2003.

2. InProtect. "Nessus & Nmap Web Interface Project". URL: http://www.inprotect.com

3. Nessus. "Documentation". URL: http://www.nessus.org/documentation.html

4. Gula, Ron. "Dedicated and Distributed Vulnerability Management." Tenable Network Security. June 2003. URL: http://www.tenablesecurity.com/distributed.pdf

5. NessusWeb. "Documentation". URL: http://www.cse.sc.edu/~chen7/NessusWeb/documentation.htm

6. SecurityNerds. URL: http://www.securitynerds.org/