Global Information Assurance Certification Paper

Securing by Effective Auditing

Dusty Hall
11-11-2003

GSEC Practical
Version 1.4b - Option 2 (Case Study)

Abstract

With close to ten thousand hosts on our network and the increasing number of vulnerabilities found each week it has proven to be impossible to physically visit each host on our network and perform a system audit, thus we turned to performing network audits. By auditing systems over the network we are able to greatly reduce the amount of time necessary to determine if a system is lacking operating system/security patches or if it is already compromised. These network audits have become a necessity for maintaining the security of our large network.

The purpose of this paper is to explain how I was able to effectively decrease the number of vulnerable systems on our network. In this paper I will focus around two Open Source tools, Nessus (http://www.nessus.org/) and Inprotect (http://www.inprotect.com/). With the use of these two tools I was able to find and eliminate vulnerable systems on our network quickly and effectively.

Before

As most people are aware desktop systems have become prime targets for malicious code. This malicious code often called Malware includes all types of Viruses, Trojans and Worms. The number of these malicious programs increases each week and every time a new vulnerability is found. The damage they cause ranges from the theft of information to denial of service attacks. All of these different types of Malware introduce security threats upon our network. An example of these vulnerabilities can be found at the following URL: http://www.microsoft.com/security/security_bulletins. This has prompted me to develop a system to reduce these threats.

Before our implementation of scheduled network audits we had no idea of how vulnerable our clients were to malicious attacks. With roughly ten thousand hosts on our network a system could be vulnerable for weeks at a time and we had no idea until it was too late and the system was compromised. This proved to be my main motivating factor for improving our network security.

Our network consists of both lab and personal computers scattered across our campus. These systems are supported by the appropriate departmental IT support staff. We heavily rely on this support staff to keep their departments computer systems up to date with the latest operating system and security patches. We also convey any relevant security issues through them for correcting any security problems in their department. Although they keep most of their clients up to date there always seems to be some that slip through the cracks. When this vulnerable system is compromised our network is introduced to a new threat. By essentially finding and patching these systems before they are compromised is the logic behind my idea of network auditing.

Due to the various different paths into our network, clients are exposed to additional threats. These paths include: email, wireless clients, VPN clients and laptops. With these different paths into our network we know we have to keep all systems patched to prevent compromises. This was very apparent from the MS Blaster Worm outbreak on our campus. Once an infected laptop connected to our network the Worm began propagating. We became quickly aware that our desktop systems were not being patched as we previously thought. This is another example that proves all clients are exposed to attacks no matter the location and that they must keep their systems routinely updated. "Worms on these systems are like rats scurrying in the gutter. They remain undetected and able to propagate for a much longer period of time." (Carr, p46). These infected systems reeked havoc propagating across our network infecting other systems and ultimately denying users the ability to perform their job function.

From the beginning I knew that network audits would provide us with a way to effectively monitor and audit systems on our network without the hassle of visiting each individual machine. My previous experience with Nessus proved to be very beneficial although it wasn't enough. Lessons learned at the GSEC course gave me new interests in this project and how I approached this issue. Being introduced to new ideas and tools I was given the additional understanding of "Security Essentials".

My prior experience with using Nessus to audit our network was focused around the problem of managing the data produced from the scans. Simple one system scans were not a problem since it was just one report but when I wanted to scan several subnets at a time the data produced was unmanageable. This data that the scans produced was good but I had no means to examine all of it easily. Another problem with my previous attempts was that I did not have any organization and all too often systems scans were overlapped. Also, I had to manually schedule each job to run which proved to be very time intensive. Any changes that I needed to make on these scans required a lot of my time and were often forgotten. With these goals in mind I started searching for the right tools for our environment.

As stated earlier the major goal of this project was to improve network security by reducing the number of vulnerable systems on our campus. The main reason I felt that it was necessary for us to perform regular scheduled network audits on our network clients was mainly due to the risks associated with the exploited vulnerabilities. "All organizations must perform complete risk assessments and implement adequate internal controls to help manage all significant risks. The need to do so has always existed, but the urgency has increased dramatically." (Champlain, xii *Preface*).

I found that there are several classifications for risk that need to be considered. CERT Coordination Center classifies risk into these three categories:

- Confidentiality
- Integrity
- Availability

Confidentiality risk is defined as information that is available only to those who rightfully have access to it. Integrity risk is information that is only to be modified by those who are authorized to do so. One common example of both of these that I have seen on our network is a Trojan allowing connections to the "C:\" allowing anyone to connect with the ability to view or delete any files on that system. Another example of this would be a system with a blank "Administrator" password. Availability risk is information that should be accessible to those who need it when they need it. An obvious example of this would be any type of denial of service attack.

All of these risks coincide directly to the effects of vulnerabilities found in systems on our network. Almost all Viruses, Trojans or Worms violate at lease one of these risks, sometimes all. Although there are several different approaches to preventing these risks from being exploited I have determined the best choice for our environment to be network audits.

During

As one of the Network Security Specialists for my organization my job is to monitor and improve the security of our network. One minor problem I faced when developing a plan to improve the security of our network was focusing on a specific area. After thinking about some recent problems we faced with a number of desktop client vulnerabilities and compromises, the choice was relatively easy. Compromised desktop clients introduce new risks for both the network and other clients. Clients infected with Viruses, Trojans or Worms can affect all aspects of a network thus affecting the clients on the network. I quickly decided to develop a system to improve the security of our network focusing on these clients.

The main approach I took in setting up this system was all based upon reducing the number of vulnerable systems on our campus. The development of a network auditing system provides a proactive means to thwart attacks on our network.

My major goals of this project can be best described by the following topics:

- Finding vulnerable systems in a quick and efficient manor.
- Ensuring that hosts are scanned on a regular basis.
- Providing IT Support Professional's with accurate and timely information regarding their department's computer security.
- Provide historical data on previous network audits.

Since I now knew my goals for this project I had to find the necessary tools.

Due to our tight budget I had to limit my choices of to Open Source or freely available applications. This ruled out applications like Internet Security System's Internet Scanner (http://www.iss.net/products_services/enterprise_protection/vulnerability_assessment/scanner_internet.php) since it was out of our price range. As mentioned earlier I had some prior experience with network auditing with Nessus and knew its capabilities. Although there were several Open Source applications that did vulnerability scanning, none came close to Nessus.

The benefits that I found in Nessus included:

- Provided detailed reports.
- Large vulnerability signature database.
- Widely accepted.
- Open Source.
- Easy to setup.

Although the benefits of network auditing out weigh the negative impact the following risks must be considered:

- Some systems may crash during scans.
- Triggers alerts on Intrusion Detection systems.
- Increases network traffic.

Since I had now decided on a vulnerability scanner the only problem left to solve was organizing the data that was produced from the scan into something that was easily readable. Scanning a single system and viewing the results was easy, but since I wanted to scan several subnets at a time the resulting data proved to be too much. This forced me to search for some type of GUI front-end that also had database capabilities.

After searching the Internet for hours I really couldn't find exactly what I was looking for in a Nessus front-end. I then decided I would attempt to write my own PHP front-end with a database back-end. I started programming and combining my ideas on what I was looking for in a front-end. After programming for a couple of weeks I accomplished a lot. I had a PHP front-end and a database setup to keep up with scans and to collect the necessary data but I was far from

completing the project. While looking over the Nessus web site one day I ran upon Inprotect. I downloaded and installed the application and started testing. I quickly realized that this was what I had been looking for and knew that it would help solve my problem.

Inprotect is an application that is based around Nessus. It includes PHP and Perl scripts for managing Nessus scans plus SQL scripts for creating a database to store all of the necessary scan information. This proved to be the finishing touch needed for my project.

Some of the benefits of Inprotect include:

- Secure PHP GUI front-end.
- Multi-user capability.
- Allows scheduling of scans.
- Real-time scan capability.
- Uses a database to manage scan information.
- Provides historical data.
- Open Source

With the above tools now chosen I was able to bring everything together to setup the system.

Since this system was created around a tight budget all tools that I have used are Open Source. The operating system I have chosen to use is Red Hat Linux 9.0 (http://www.redhat.com/) operating on an Intel platform. The main reason I have chosen Red Hat is because of its wide acceptance and freely available. Although I have chosen to install this system on the Red Hat distribution of Linux it should work with any recent Linux distribution.

Due to the constraints of this paper I will assume that the reader has had prior experience with the Linux/Unix operating system and is familiar with installing open source software. I also expect that the user will already have the latest stable versions of the following applications installed:

- Apache httpd (http://httpd.apache.org/)
- PHP (http://www.php.net/)
- MySQL (http://www.mysql.com/)
- Nmap (http://www.insecure.org/)

All of these applications should come installed by default on most installations of Red Hat 9.0. If not most of these applications are easily installed from RPM's directly downloaded from Red Hat's website, URL: (http://www.redhat.com/apps/download/results.html).

The combining of all of these applications provides a robust network auditing system. Before I start with the installation I would like to give an overview of how

this system operates. As mentioned earlier Nessus is a vulnerability scanner and Inprotect is a GUI interface to Nessus. Inprotect is a web application written in PHP that lets you manage all aspects of Nessus. Inprotect provides everything from scheduling to viewing scan reports from a web page. Also included with the Inprotect package is a set of Perl scripts. These scripts execute the scans scheduled by the PHP and insert data into the database. All data is stored in a database so that the scan data and scheduling information is easily stored and retrieved.

To begin the setup of the network auditing system we will need to install and setup Nessus. The Nessus installer script (nessus-installer.sh) can be downloaded at the following URL: http://www.nessus.org/. To start the installation, run the following script:

./nessus-installer.sh

Once this script is finished running Nessus should be installed. You will now need to do the following:

- Create a nessusd certificate: /usr/local/sbin/nessus-mkcert
- Add a nessusd user: /usr/local/sbin/nessus-adduser
- Start the Nessus daemon: /usr/local/sbin/nessusd –D

If you are experiencing any problems with the installation of Nessus I would suggest that you visit the following URL:
http://www.linuxsecurity.com/feature_stories/nessusintro-printer.html
This site has a very good tutorial on the installation and setup of Nessus from start to finish.

Several Perl Modules need to be installed before we can install Inprotect. These modules include:

- DBI
- MIME::Lite
- Parallel::ForkManager
- Date::Calc

These Perl Modules can be installed by issuing the following command at the command line:

perl –MCPAN –e shell
cpan> install <*module::name*>

More information regarding the installation of Perl Modules using CPAN can be found at URL: http://www.perl.com/doc/manual/html/lib/CPAN.html

Next we will need to download Inprotect from the following URL: http://sourceforge.net/projects/inprotect. After downloading you will need to extract the tarball into a temporary folder.

tar –xzvf inprotect_014.tar.gz

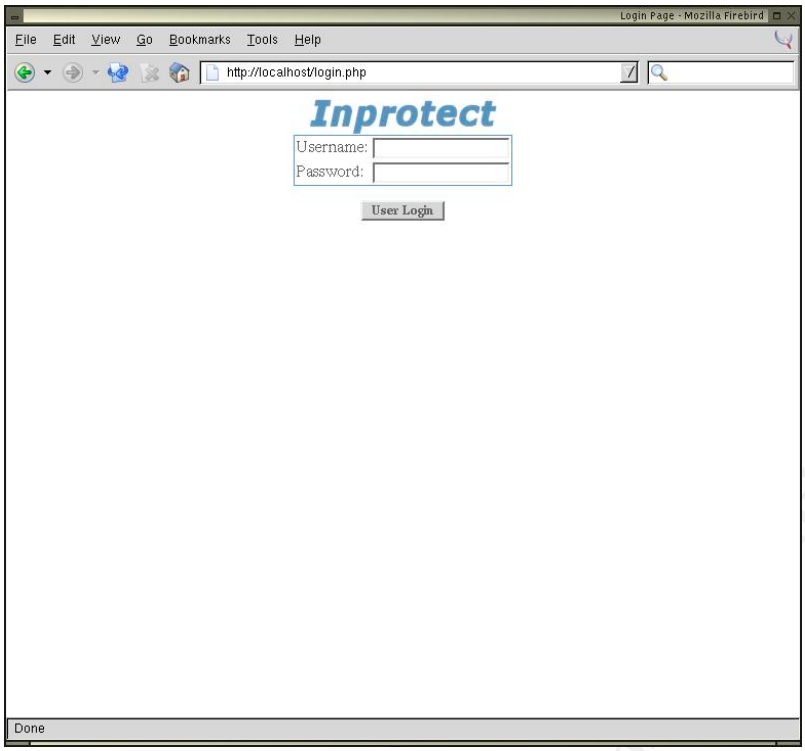
Included with this tarball is a Red Hat installer script that installs the necessary files automatically. Simply run this command from the Inprotect directory:

./rh_installer.sh

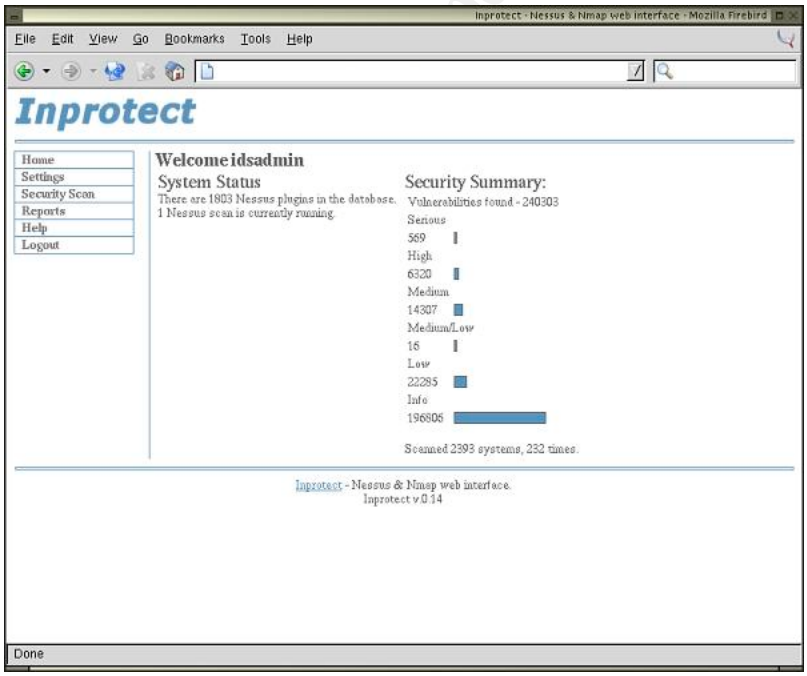
After this is completed it is necessary to customize several configuration files. The first of these is /var/www/html/config.php, this file contains information so that the Inprotect PHP can communicate with the MySQL database. You will need to edit this file and modify the user and password to match your MySQL setup. Another file is /usr/local/bin/inprotect.cfg, this file contains the information necessary to connect to the Nessus daemon. Edit this file and modify the information relevant to your Nessus installation.

I tried to provide enough information necessary to ease the installation. Please consult the README file provided in the Inprotect tarball regarding the detailed instructions for setup and installation.

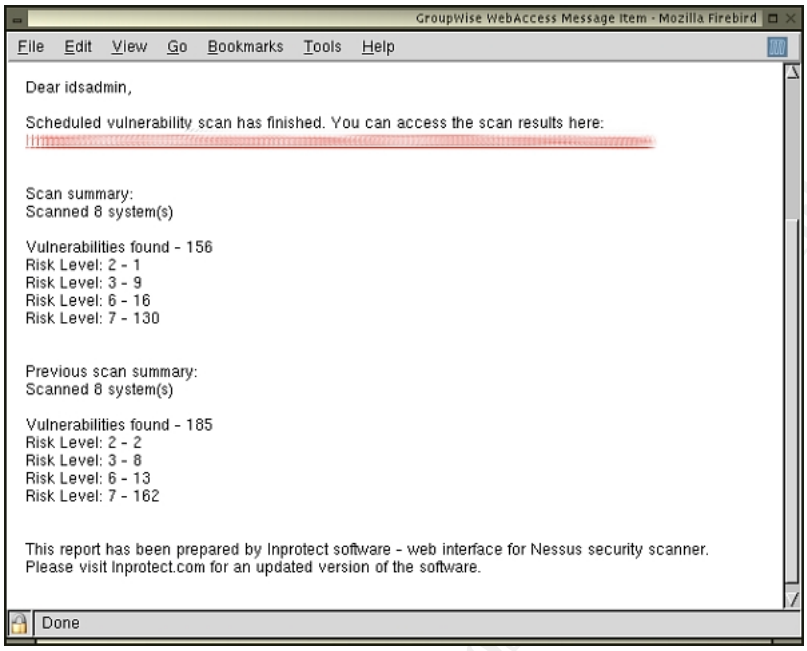
With the necessary PHP files in place we can now login Inprotect. Open up a web browser and point it to the hostname where Inprotect is installed. You will be prompted for a username and a password. The default username is admin and the password is password.



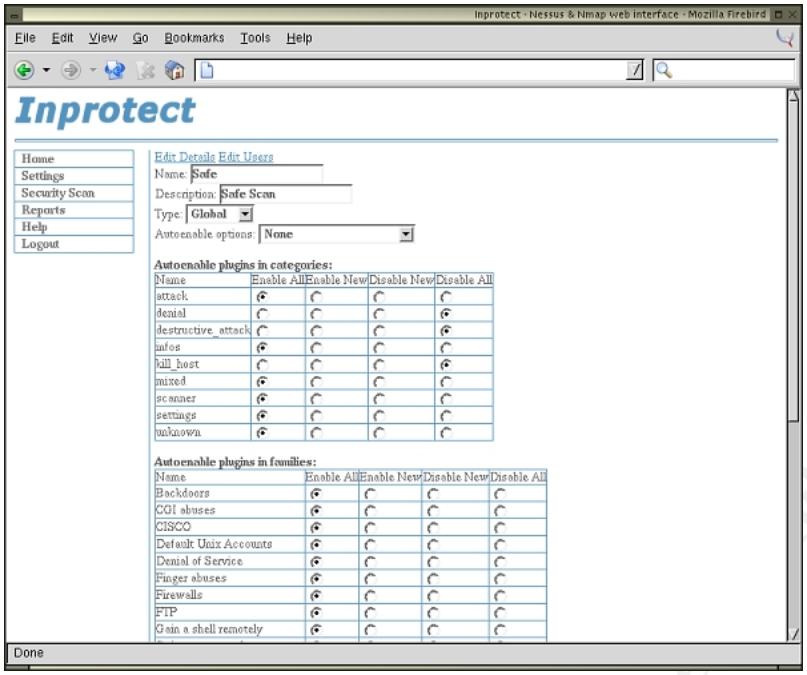
Once logged into Inprotect you are now well on your way to begin auditing your network. Several different aspects will need to be configured first before a network audit can be performed. Please remember the risked associated with performing network audits as mentioned earlier in this paper.



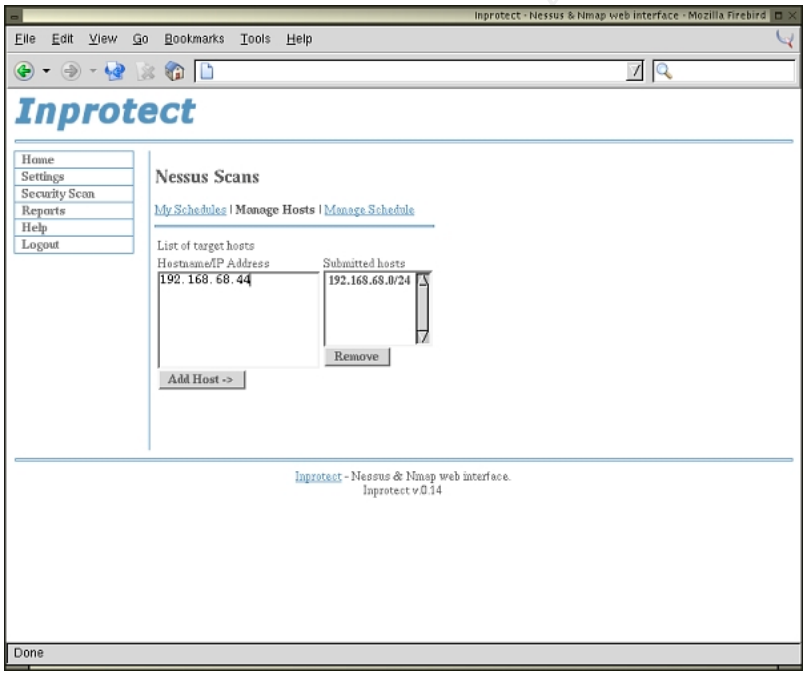
To begin with I would advise changing the default password. This is all manageable from the settings link on the left of the page. After changing the initial password it is necessary to update the email address associated with the account. By entering a valid email address here scan starting times, ending times and results summaries will be emailed for each scan. This provides you with a quick summary of all your scans.



The next step we must take before we can scan is to setup a scan profile. The scan profile is a set of values that you would like Nessus to use during its scans. These are very customizable, for example I have a scan profile just for determining if a user has a blank password. Also, on Inprotect's website (http://www.inprotect.com/download/top20.cfg) they have a "SANS Top 20" scan profile which checks for the top 20 most common vulnerabilities published by SANS. With profiles this easily configurable it is adaptable to any network.

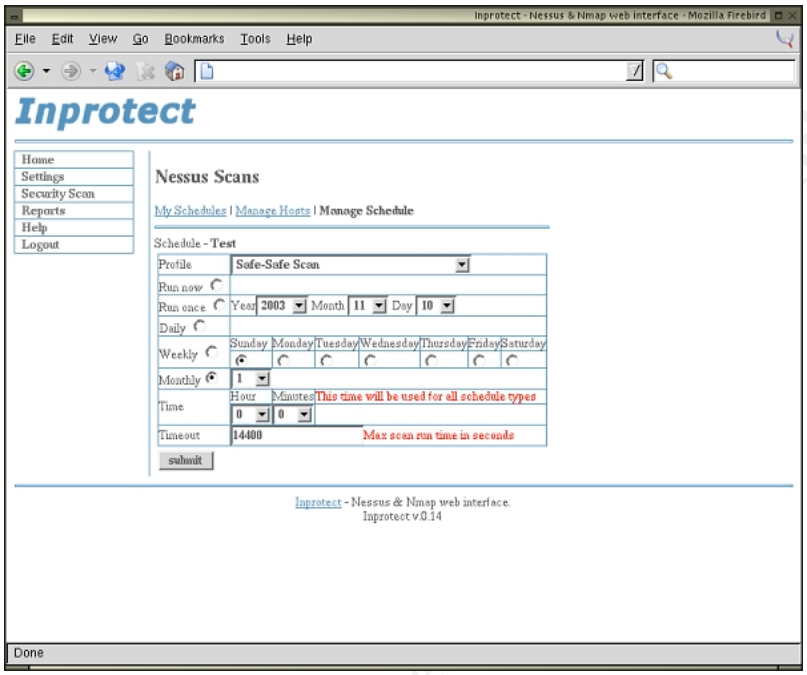


After the needed profile is created we can now setup our first scan. In order to do this we must use the "Security Scan" link on the left and then chose "Nessus Scan". We are then prompted for a name for our scan, insert a simple description. Now it is necessary to insert the host(s) that we would like to scan.



Once the hosts are inserted we can now schedule our scan. There are several different options available for scanning, these options include:

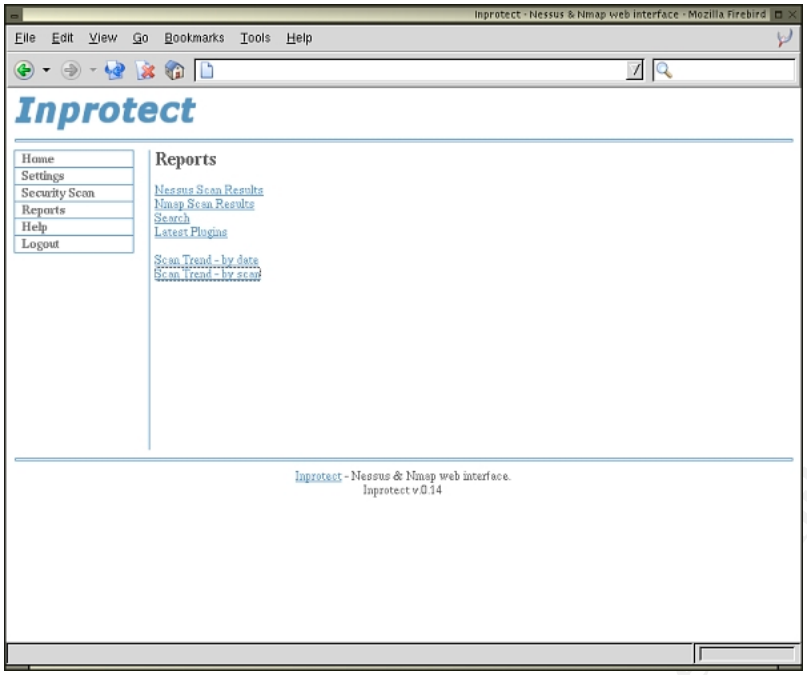
- Now
- Once
- Daily
- Weekly
- Monthly



After the desired time is selected and the submit button is activated the schedule is entered into the queue for executing. This is viewable from the "View running scans details" link at the bottom of the "Nessus Scans" page. Percent meters are displayed to give you an approximate ending time. Average scans take around 15 to 20 minutes and depend on network, host and client speed.

When the scan is finished we are now able to view the results. These are viewable through the "Reports" link. Inprotect provides several different types of reports that include:

- Nessus Scan Results
- Nmap Scan Results
- Search
- Scan Trend - by date
- Scan Trend - by scan



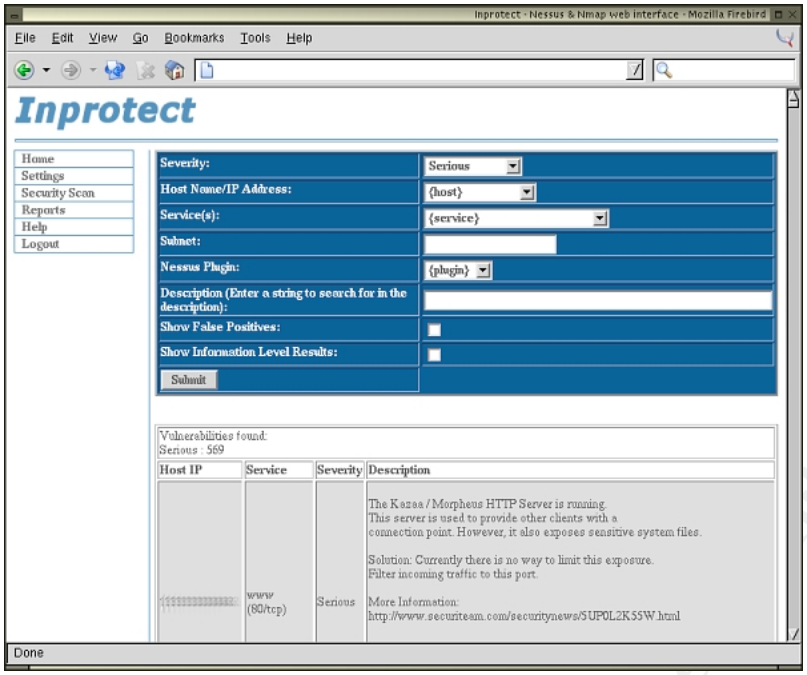
The different types of reports offered by Inprotect played a huge part in persuading me to use Inprotect. Since everything is stored in a database scan data is easily organized and searchable.

By now you should have a good idea of what to expect from the combined benefits of Nessus and Inprotect. Although Inprotect is still in its early stages of development, version 0.14 at time of writing, it definitely did not fall short of any of my expectations. Harnessing the ability to perform scheduled network audits efficiently and effectively was definitely carried out through the setup of this system.

After

The setup of this network auditing system has definitely improved the security of our campus network. The benefits of our network audits have been very evident through the decrease in security incidents since the rollout. Obviously there are still vulnerable areas on our network and this is not the end all solution to network security but it has definitely taken us to the next level. Currently some of the benefits we see include:

- Vulnerable systems are quickly found.
- Scheduled scans provide consistency.
- Custom scanning profiles.
- New vulnerabilities are caught quickly.
- Historical data of previous scans.



This system was easily setup in one business day although the tuning and maintenance take longer to configure and depend on the size of the network. Our system has been in place for approximately three months and requires little maintenance other than adding new systems. I login to the system in the morning and view the previous nights scan and take the necessary action for any problems I find.

An easily noticeable effect from these scans can be seen in the number of compromises that were found on our network. Before I would typically find 1-2 compromised systems a day, now we see less than that for an entire month. We found that most of these systems had blank Administrator passwords. Creating a custom profile allowed me to perform quick network scans to find hosts with blank Administrator passwords. Another advantage to this system is the automatic updating of the Nessus vulnerability database. This provides us with up to date signatures on a daily basis. Since vulnerability patches are usually made available before exploits, we are able to find and get these systems patched before they are compromised.

As mentioned earlier in this document Nessus can cause problems on some systems that it scans and you should use caution when scanning new systems. This could be the crashing of an application or the shutdown of the entire system. I experienced one such problem with RPC services on a few SUN Sparc systems. When Nessus performed a “Safe” audit of this system it caused the RPC services to crash. This forced me to stray from scanning those hosts. Some other problems that I had to deal with included “False Positives”. Nessus is not extinct from detecting vulnerabilities that do not exist. After looking through the data produced for some time I can quickly spot most false positives.

Throughout this project I was able to use both my existing knowledge and information learned at SANS training. My prior knowledge of open source tools was essential in researching and setting up the system. Information learned at SANS helped me in my understanding of security concepts and identifying problem areas in network security. I am very proud of this system and of the information that it provides us with. It has saved me a lot of time and grief in my quest of removing vulnerable clients from our network. Although the setup of this system is relatively easy, the combined benefits of these tools create a very robust system that helps keep our network secure.

References

Microsoft Corporation. “Microsoft Security Bulletins.”: October 15, 2003. URL: http://www.microsoft.com/security/security_bulletins

Carr, Jim. “Preventing Internal Attacks.” Network Magazine September 2002 (2002): 46.

Champlain, Jack. Auditing Information Systems. Hoboken: John Wiley & Sons, 2003. xii *Preface*.

CERT Coordination Center. “Home Network Security.”: June 22, 2001. URL: http://www.cert.org/tech_tips/home_networks.html (December 5, 2001)

Inprotect.com. “Inprotect Project.”: November 01, 2003. URL: http://sourceforge.net/projects/inprotect

Nessus.org. “Nessus Documentation.”: October 23, 2003. URL: http://www.nessus.org/documentation.html

Harangsri, Banchong. “Introduction to Nessus, a Vulnerability Scanner.”: July 07, 2002. URL: http://www.linuxsecurity.com/feature_stories/nessusintro-printer.html

König, Andreas. “CPAN.”. URL: http://www.perl.com/doc/manual/html/lib/CPAN.html

Last Updated: October 17th, 2017

Upcoming Training

SANS vLive - SEC401: Security Essentials Bootcamp Style	SEC401 - 201710,	Oct 23, 2017 - Nov 29, 2017	vLive
SANS San Diego 2017	San Diego, CA	Oct 30, 2017 - Nov 04, 2017	Live Event
San Diego Fall 2017 - SEC401: Security Essentials Bootcamp Style	San Diego, CA	Oct 30, 2017 - Nov 04, 2017	vLive
SANS Seattle 2017	Seattle, WA	Oct 30, 2017 - Nov 04, 2017	Live Event
SANS Gulf Region 2017	Dubai, United Arab Emirates	Nov 04, 2017 - Nov 16, 2017	Live Event
Community SANS Vancouver SEC401^	Vancouver, BC	Nov 06, 2017 - Nov 11, 2017	Community SANS
Community SANS Colorado Springs SEC401~	Colorado Springs, CO	Nov 06, 2017 - Nov 11, 2017	Community SANS
SANS Miami 2017	Miami, FL	Nov 06, 2017 - Nov 11, 2017	Live Event
SANS Sydney 2017	Sydney, Australia	Nov 13, 2017 - Nov 25, 2017	Live Event
SANS Paris November 2017	Paris, France	Nov 13, 2017 - Nov 18, 2017	Live Event
SANS San Francisco Winter 2017	San Francisco, CA	Nov 27, 2017 - Dec 02, 2017	Live Event
Community SANS St. Louis SEC401	St Louis, MO	Nov 27, 2017 - Dec 02, 2017	Community SANS
SANS London November 2017	London, United Kingdom	Nov 27, 2017 - Dec 02, 2017	Live Event
Community SANS Portland SEC401	Portland, OR	Nov 27, 2017 - Dec 02, 2017	Community SANS
SANS Khobar 2017	Khobar, Saudi Arabia	Dec 02, 2017 - Dec 07, 2017	Live Event
SANS Austin Winter 2017	Austin, TX	Dec 04, 2017 - Dec 09, 2017	Live Event
SANS Munich December 2017	Munich, Germany	Dec 04, 2017 - Dec 09, 2017	Live Event
Community SANS Ottawa SEC401	Ottawa, ON	Dec 04, 2017 - Dec 09, 2017	Community SANS
SANS vLive - SEC401: Security Essentials Bootcamp Style	SEC401 - 201712,	Dec 11, 2017 - Jan 24, 2018	vLive
SANS Bangalore 2017	Bangalore, India	Dec 11, 2017 - Dec 16, 2017	Live Event
SANS Cyber Defense Initiative 2017	Washington, DC	Dec 12, 2017 - Dec 19, 2017	Live Event
SANS Cyber Defense Initiative 2017 - SEC401: Security Essentials Bootcamp Style	Washington, DC	Dec 14, 2017 - Dec 19, 2017	vLive
Community SANS Hawaii SEC401	Honolulu, HI	Jan 08, 2018 - Jan 13, 2018	Community SANS
Community SANS Nashville SEC401^	Nashville, TN	Jan 08, 2018 - Jan 13, 2018	Community SANS
SANS Security East 2018	New Orleans, LA	Jan 08, 2018 - Jan 13, 2018	Live Event
Mentor Session - SEC401	Memphis, TN	Jan 09, 2018 - Mar 13, 2018	Mentor
Northern VA Winter - Reston 2018	Reston, VA	Jan 15, 2018 - Jan 20, 2018	Live Event
SANS Amsterdam January 2018	Amsterdam, Netherlands	Jan 15, 2018 - Jan 20, 2018	Live Event
Mentor Session - SEC401	Minneapolis, MN	Jan 16, 2018 - Feb 27, 2018	Mentor
SANS Las Vegas 2018	Las Vegas, NV	Jan 28, 2018 - Feb 02, 2018	Live Event
Las Vegas 2018 - SEC401: Security Essentials Bootcamp Style	Las Vegas, NV	Jan 28, 2018 - Feb 02, 2018	vLive