
This is the **published version** of the bachelor thesis:

Piqué Mañas, Arnau; Pons Pons, Pedro Luis, dir. Systematic use of tools for detecting computer vulnerabilities. 2022. (1395 Grau en Gestió de Ciutats Intel·ligents i Sostenibles)

This version is available at <https://ddd.uab.cat/record/264108>

under the terms of the  license



Universitat Autònoma
de Barcelona

Grau en gestió de ciutats intel·ligents i sostenibles

SYSTEMATIC USE OF TOOLS FOR DETECTING COMPUTER VULNERABILITIES

Creation of a series of vulnerability scanners, which allow
for increased security and are easy for users to use

Author: Arnau Piqué Mañas

Tutor: Pere Lluís Pons Pons

Arnau.PiqueM@autonoma.cat

INDEX

1 Introduction.....	1
2 Methodology	2
2.1 Scanners configuration on OpenVAS	2
2.2 Target configuration on OpenVAS	2
2.3 Task creation	4
2.3.1 Alert creation.....	4
2.3.2 Schedule creation	5
2.3.3 Performance configuration	5
3 Differences in the creation of scans between OpenVAS and Nessus	7
4 Creation of scanners in Nessus	7
4.1 Discovery scan	7
4.2 Malware scan.....	8
4.3 Credentialed patch scan	8
4.4 Vulnerability scans.....	9
4.4.1 Basic scan.....	10
4.4.2 Solorigate scan.....	10
4.4.3 ProxyLogon: MS Exchange scan	11
4.4.4 CISA Alerts AA22-01 and AA22-047A scan.....	11
4.4.5 ContiLeaks scan	11
4.4.6 PrintNightmare scan.....	11
5 Implementation of the developed scanners.....	12
5.1 Nessus scans results.....	12
5.1.1 Discovery Scan results	12
5.1.2 Basic scan results.....	13
5.1.3 WEB scan results	13
5.1.4 MS Exchange scan results	14
5.1.5 ContiLeaks scan results	14
5.1.6 Log4Shell results	15
5.1.7 Solorigate results.....	15

5.1.8 Cisa scan results	15
5.2 OpenVAS scans results	16
5.2.1 Discovery scan results	16
5.2.2 Log4shell scan results	17
5.2.3 Full and fast scan results	17
5.2.4 Full scan results	18
6 Conclusion	18
7 Agreements	19
8 References	19

Abstract

In this final degree project, a systematic set of tools for scanning application vulnerabilities has been created. two programmes with free version were chosen for this purpose: OpenVAS and Nessus. Once all the scanners of each respective programme had been created, they were tested on two applications under development at the Autonomous University of Barcelona, and an analysis of the vulnerabilities found was carried out.

Resumen

En este proyecto final de carrera, se ha creado una sistemática de herramientas para escanear las vulnerabilidades de aplicaciones. Para ello se han elegido dos programas con versiones gratuitas: OpenVAS y Nessus. Una vez creados todos los escáneres de cada respectivo programa, se han puesto a prueba en dos aplicaciones en desarrollo de la Universidad Autónoma de Barcelona, y se ha hecho un análisis de las vulnerabilidades encontradas

Key words

Scan, vulnerability, TCP port, UDP port and server

1 Introduction

Nowadays we live in a world where technology plays a very important role in our society, there are more and more people with electronic devices, such as computers or smart phones. We can also observe how the number of these devices has increased in the use of infrastructures, companies, and services. As the number of users increases, the number of cyber-attacks is also growing, especially since the coronavirus pandemic where many businesses opted to go digital. These attacks occur daily, with small and medium-sized enterprises suffering the most, according to various studies,

In the United States, according to Bryan Watkins, "National Security Agency (NSA) Director General Keith Alexander referred to cyber espionage as "the greatest transfer of wealth in history". Globally, the cost of cybercrime is estimated at more than \$385 billion" (Watkins, 2014).

As we can see, the cost of cyber-attacks is very high and usually there is no solution or if there is a solution it is usually quite expensive, and the process can paralyse the company or institution for weeks or months. Therefore, the best way to protect yourself is to have secure systems. To achieve this, the system must be analysed, and a study of its vulnerabilities must be carried out. In this final degree project, I propose the creation of a system of vulnerability

scanner tools to help us improve the security of our computer equipment, which are easy to use by the user, without the need to have a great prior knowledge about its operation.

2 Methodology

The operating system chosen for this project is Kali Linux, specifically version 2022.1, as it is the most recent and stable version. Instead of installing Kali Linux in a partition of my computer, we will install it as a virtual machine, for 2 reasons. The first is that a virtual machine has a very simple installation process and the second is that the virtual machine allows us to make copies, so we can generate backups of the virtual machine in case there is any problem that prevents its operation. The programs chosen to create the tools are OpenVAS, version 21.4.4, and Nessus, version 10.2.0, programs very present in the cybersecurity field. Once I have decided the software I am going to use, I will go on to the creation of the scanners, for this I will have to study the manuals of the OpenVAS and Nessus programs, as the aim of this work is to create systematic use of tools to detect vulnerabilities, I will not only have to study the parameters for the creation of scanners, but I must also find out how to create it, so that it can be used by any user. Once I have the scanners created, they will be tested on 2 servers. The parameters that will be followed to compare the results of the scanners will be the number of vulnerabilities detected and the execution time.

2.1 Scanners configuration on OpenVAS

Next, we will look at the process for creating scanners in OpenVAS. This creating process is divided in 2 parts, one is the target, the server to study and the other the scanner itself.

2.2 Target configuration on OpenVAS

For the configuration of the target, I must first decide which ports are going to be scanned. For the configuration of the target, in my case I have decided to choose the option All IANA assigned TCP and UP, as this way as many ports as possible will be scanned. In this way the result obtained will be better, as a larger number of ports have been analysed.

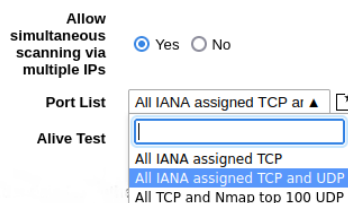


Figure 1 -target ports configuration

In the Alive Test field, we use it to determine some settings of the ping scanner, as we can see in the image, we have different options. OpenVAS gives us the default option Scan Config

Default. In my case I have chosen the Scan Config Default option as it is the most complete one.

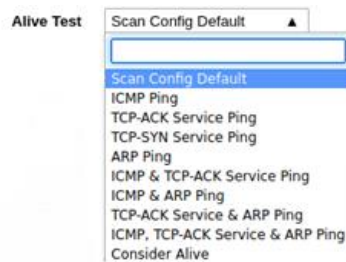


Figure 2- Alive test

If we continue exploring the options that exist in OpenVAS when configuring the target, we find the option credentials for authenticated checks. OpenVAS offers us but different credentials, but to analyse our target we will only perform the SSH and SMB credentials.

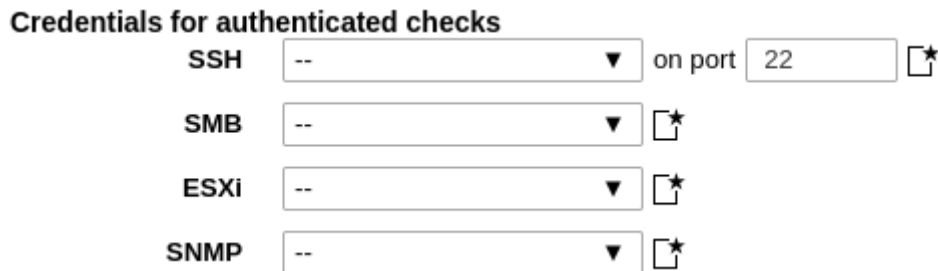


Figure 3 - credentials OpenVAS

“SSH or Secure Shell is a network communication protocol that enables two computers to communicate (c.f http or hypertext transfer protocol, which is the protocol used to transfer hypertext such as web pages) and share data. An inherent feature of ssh is that the communication between the two computers is encrypted meaning that it is suitable for use on insecure networks.

SSH is often used to "login" and perform operations on remote computers, but it may also be used for transferring data (London, 2022)”

SMB windows server, usually use 445 port, but although can use TCP port 139 and UDP port 137 and 138.

“The Server Message Block (SMB) protocol is a network file sharing protocol that allows applications on a computer to read and write to files and to request services from server programs in a computer network. The SMB protocol can be used on top of its TCP/IP protocol or other network protocols. Using the SMB protocol, an application (or the user of an

application) can access files or other resources at a remote server. This allows applications to read, create, and update files on the remote server. SMB can also communicate with any server program that is set up to receive an SMB client request. (Microsoft, 2022)”

These credentials are very useful to create the scanners, as they allow a deeper analysis of the vulnerabilities of the system to be analysed. If we create a scanner without credentials, the vulnerability analysis will be only the vulnerabilities detected through the network, we could say that it would be like performing a superficial scan, whereas, if we include the credentials in the scanner, the scanner we will have will be more precise since, with these credentials, the Open Vas programme will be able to analyse the vulnerabilities in a deeper way, as it will be able to perform an analysis from inside the system. It allows us to identify the weakest configurations and the patches that have yet to be applied to the system, in addition to finding a greater number of vulnerabilities. Although the creation of credentials can be very beneficial when it comes to implementing it in our scanner, I think it is important to mention that it also has several negative points. Scanning with credentials can lead to the interruption of the functions of the system we are analysing. It should be noted that, in Open Vas, as we can see in the image below, it offers the option of using credentials without interrupting the functionality of the system.

For the creation of my scanner, I have decided to allow credentials to interrupt the target functions, because with this we get a deeper analysis and after studying the OpenVAS manual and different web pages, interrupting the target functions is something difficult to happen.

2.3 Task creation

In this section I have decided to include the creation of alerts and schedules. This is because one of the objectives of this project is that the scanners created are easy to use by users, with the creation of these sections we get the scanners to run and send the results automatically, so that the user does not have to intervene in the process.

2.3.1 Alert creation

OpenVAS gives us the option to send reports by different methods. In my case, I have configured it to be sent whenever the alert is executed, once the scan is finished, regardless of whether new vulnerabilities have been detected. The method chosen to send the alert has been via email, as I consider it a simple method to send and receive the reports. I have decided not to encrypt the information as I don't think it is necessary. Finally, I have included the report in txt format so that the email contains all the information.

Method: Email

To Address:

From Address:

Subject: Report Scan

Email Encryption: --

Simple Notice ☐

Include report ☒ TXT

Task '\$n': \$e'

After the event \$e,
the following condition was met: \$c

This email escalation is configured to apply report format '\$r'.
Full details and other report formats are available on the scan engine.

Content: \$t

Figure 4 - alert configuration OpenVAS

2.3.2 Schedule creation

When configuring the task, we find the scheduling function. This OpenVAS function allows you to schedule a timetable for the scanner. In the following image I have configured a schedule, I have configured it to scan once every day without a set date. Another schedule has been created, this is for the selected scanner to be performed once a week, it is designed to perform a type of scan that can cause a malfunction of the server but at the same time perform a deeper analysis of vulnerabilities.

Name: Schedule

Comment:

Timezone: Coordinated Universal Time/UTC

First Run: 06/09/2022 23:00 Now

Run Until: 06/09/2024 00:00 Open End

Duration: Entire Operation

Recurrence: Daily

Figure 5 - schedule for the scans I

Name: schedule weekly

Comment: schedule for deep scanners

Timezone: Coordinated Universal Time/UTC

First Run: 06/09/2022 23:00 Now

Run Until: 06/10/2022 00:00 Open End

Duration: Entire Operation

Recurrence: Weekly

Figure 6 - schedule for the scans II

2.3.3 Performance configuration

The option Add results to Assets, allows the system to be available for automatic management. I have selected this option as it facilitates the scanning process. I have checked the option to apply overrides, as the function of overrides is to be able to modify the severity of the results.

Open Vas classifies vulnerabilities as high, medium, low and log. If we apply override and there is a vulnerability detected as low, but after a study I conclude that it is very dangerous, we can modify the scanner so that every time it detects the vulnerability, it shows it as high. I have applied a Min Qod of 100%, as this parameter indicates the quality of vulnerability detection. It should be noted that by applying 100% the time it will take for the scan to be performed will be longer than if we apply the default percentage, which is 70%. Alterable task allows to modify the task, even if we already have the reports created, although this creates a problem, as the coherence of the reports cannot be guaranteed, that's why I have decided not to allow this option, if we want to make other tests it will be better to create another task. I have decided not to check the Auto Delete Reports option because I think it is better to let the user decide whether to delete the reports or not.

The image shows a configuration window for OpenVAS scan parameters. It contains several settings:

- Add results to Assets:** Radio buttons for Yes (selected) and No.
- Apply Overrides:** Radio buttons for Yes (selected) and No.
- Min QoD:** A text input field containing '100' followed by a percentage sign (%).
- Alterable Task:** Radio buttons for Yes and No (selected).
- Auto Delete Reports:** Two radio buttons: 'Do not automatically delete reports' (selected) and 'Automatically delete oldest reports but always keep newest'.
- Reports:** A text input field containing the number '5' followed by the word 'reports'.

Figure 7 - OpenVAS scan parameters I

If we select OpenVAS Default as scanner, we must select the scanner configuration. My choice was to select 4 of these scanner configurations. Discovery is a scanner that allows OpenVAS to get information about open ports, firewalls, hardware, used services and certificates. Log4Shell is a scanner to detect vulnerabilities in log4j, which is an open-source library used by developers. It is quite commonly found on programmers' computers. Full and fast is a scanner configuration that allows us to do vulnerability analysis without compromising the system. The last configuration I selected was one I created myself, called Full scan. This configuration is a heavier vulnerability scan. It takes longer than the full and fast configuration and can cause problems on the system being scanned, that's why this configuration is more optimal to use when the system is down.

The last 3 sections configured are Order for target hosts, Max NVTs currently running per host and Max hosts scanned concurrently. I have selected the order of the target hosts as random, because after studying the OpenVAS manual, this option is recommended to improve the scanner estimation. As for the last two parameters, I have left the values given by default by OpenVAS, because a higher value of the parameters can generate problems in the network.



The image shows a configuration window for OpenVAS scan parameters. It contains three settings:

- Order for target hosts:** A dropdown menu currently set to 'Random'.
- Maximum concurrently executed NVTs per host:** A numeric input field set to '4'.
- Maximum concurrently scanned hosts:** A numeric input field set to '20'.

Figure 8 - OpenVAS scan parameters II

To conclude this section, the tests in OpenVAS will be 4 scans. The scanners: Full scan, Log4Shell, Full and fast and Discovery,

3 Differences in the creation of scans between OpenVAS and Nessus

Before we start moving on to the creation of the scanners in Nessus, I think it is appropriate to make a comparison between OpenVAS and Nessus. It should be noted that Nessus is a paid programme, but it has a free version, which is somewhat limited, for example it only allows us to have a scanner with a schedule, but for the practical part it will be enough. If we wanted to put this system of scanners in real life, the ideal would be to have the paid version.

Nessus offers a larger number of default scanners. These fulfil a number of specific and different functions, such as performing a scan to see if the target has all the updates or just scanning for vulnerabilities in the web pages stored on the target system. Another difference that I think is important to mention is that unlike OpenVAS, the ability to customise the target. While in OpenVAS, the target was independent of the scanner and we could customise the list of ports or the Alive Test section, in Nessus we find that each scanner has its own target and its configuration is very limited, it only allows us to set the IP address of the target.

4 Creation of scanners in Nessus

For the creation of the Nessus scanners, I have decided to divide it into 2 parts. One will be on scanners, not focused on vulnerabilities and the other will be a part entirely dedicated to vulnerability scanners.

4.1 Discovery scan

One of the most important scanners we should start with is the Discovery scanner. This scanner performs a scan on the target and takes care of finding and enumerating the live hosts and open ports. It is a scanner whose configuration is somewhat limited, in my case I have modified it to scan all ports, as by default it only scans common ports. For this scanner, I think it is necessary

to run it every day, as it gives us very important information about our target, and it is not a big burden on the network.

4.2 Malware scan

Malware scan helps us to detect if our target has any malware. This scanner is very important, because while vulnerability scanners are used to detect the security holes we have, Malware scan detects malware that may already exist on our computer. I have decided to include fragile hosts in the scanner as well, because printers and Novell NetWare hosts are also included in the scanner.

“Novell NetWare is type of Network Operating System. It provides wide networking services ranging from easy and simple file to network user, data, security, and even resource management. It is generally designed for networks or Local Area Network (LAN) operating system. (Hammad, 2022)”

Another option I have decided to activate, is to have the system files scanned. This slows down the scanner, but in return the scanner will be more accurate and increase the security. The schedule for this scan is set to be done once a week. This is because it is a scanner that is slow and can cause disruption to the target service. The schedule for this scan is set to be done once a week. This is because it is a scanner that is slow and can cause disruption to the target service.

4.3 Credentialed patch scan

This scanner scans the target and looks for missing updates. This is a key aspect to increase security, as updates to services and software usually include new functionalities and fix vulnerabilities.

One of the options that I have modified has been to activate request information about SMB Domain, this option is disabled by default, but I think it is appropriate to activate it because although the scan time is increased, we will get more information. I have selected the option to scan all ports. This causes a larger number of services to be scanned.

I have activated the Oracle Database option because this way if the target has a database. With this option Nessus will authenticate to the database with the detected SIDs. SID is the database instance through which you connect to the database. An instance will have the same name as the database but without the domain.

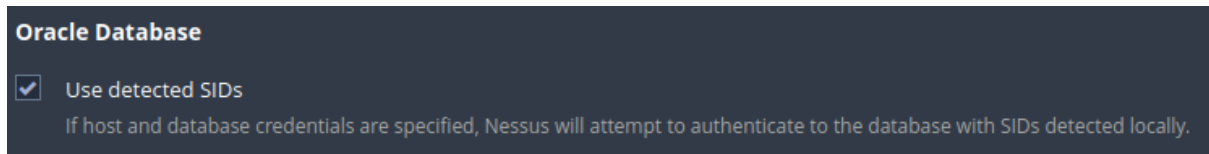


Figure 9 - Nessus Oracle Database

I have decided to set the schedule for the scanner to run on a daily basis, as it is important to keep all software up to date.

4.4 Vulnerability scans

Nessus offers several vulnerability scanners. Each of these has a different database. I have decided to list all the scanners, since each one has a different database, we will have a tool that will be able to detect a wide range of vulnerabilities. Nessus allows a rather limited configuration, the fields that can be configured in the vulnerability scanners are the same. That is why we will see below the configuration I have done for all Nessus vulnerability scanners.

I have decided to activate the Fragile Devices option, because all devices should be scanned to increase security. It could be the case that one of these devices has a vulnerability that could cause the rest of the system to be infected.

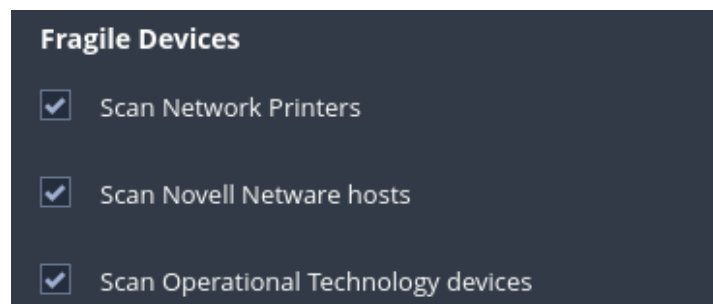


Figure 10 - Nessus Fragile devices

In the Network Port Scanners section, I have decided to activate the option Use aggressive detection in TCP and SYN, as we will get a deeper analysis. I have decided not to include UDP, as Nessus shows a message which says that it can cause unreliable results.

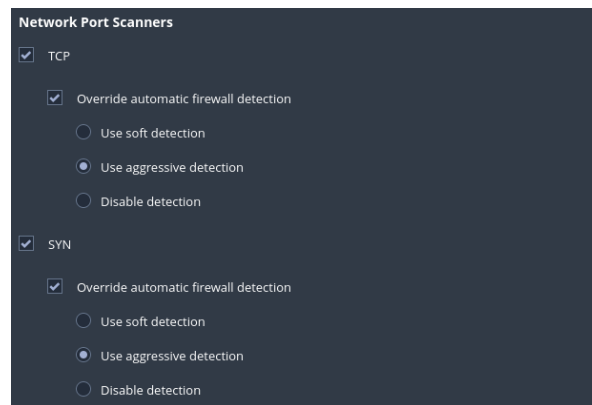


Figure 11 - Nessus network port scanners

By default, Nessus only scans the known TCP and UDP ports. I have chosen to scan all TCP and UDP ports to improve the scan result.

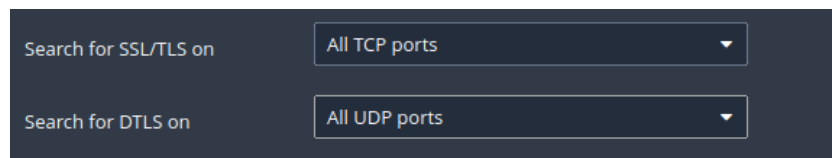


Figure 12- TCP and UDP ports configuration

I have decided to set a daily schedule, because I think it is appropriate to establish that vulnerability scans should be run on a daily basis, to increase the security of the target.

4.4.1 Basic scan

Basic network scan is a general vulnerability scanner. While the other scanners focus on very specific vulnerabilities, this scanner will scan all types of vulnerabilities found in the Nessus database.

4.4.2 Solorigate scan

Solorigate scan is a scanner that detects SolarWinds Solorigate vulnerabilities.

“SolarWinds is a major software company based in Tulsa, Okla., which provides system management tools for network and infrastructure monitoring, and other technical services to hundreds of thousands of organizations around the world. Among the company's products is an IT performance monitoring system called Orion. (Saheed Oladimeji, 2021)”

“A Solorigate is an attack that took place on the SolarWinds software. Technically, the attacker got into a remote management software server of the company and somehow

injected a backdoor into the software update of Orion. And as soon as the SolarWinds started pushing the update to its customers, the backdoor induced software was deployed onto the customer systems- thus letting the hackers sniff out the operations being carried out of the systems loaded with the software update; leading to espionage. (Goud, 2022)”

4.4.3 ProxyLogon: MS Exchange scan

This scan detect Exchange vulnerabilities targeted by HAFNIUM.

“Vulnerabilities in Microsoft Exchange servers allowed hackers to access a company’s servers, emails, and calendars. Hafnium, a group of hackers that is well trained and operates in a sophisticated manner from China is the culprit. The Hafnium Hack (connect, 2021)”

4.4.4 CISA Alerts AA22-01 and AA22-047A scan

This scanner is designed to detect vulnerabilities used by Russian-sponsored hacking groups. AttackIQ has developed a new attack graph to help organizations test and validate their cyberdefenses against known Russian adversarial tactics, techniques, and procedures

4.4.5 ContiLeaks scan

This scanner looks for vulnerabilities found in ContiLeaks chats. These chats were formed by hackers, who shared information about vulnerabilities, luckily, these chats have been leaked and a database of vulnerabilities named in these chats has been created.

4.4.6 PrintNightmare scan

PrintNightmare scan is a scanner that detects vulnerabilities affecting the Microsoft Windows Print Spooler Service. This service has the function of saving the files in the printer queue on a computer.

“this vulnerability has the potential to enable cyber-attackers to gain complete control of an affected system.As the Print Spooler service is run on Domain Controllers, an attacker could insert DLLs into a remote Windows host, whereby a regular domain user can execute code as SYSTEM on the Domain Controller.This vulnerability has now

been weaponised, exploit code exists in the wild, and it has been incorporated into popular post-exploitation frameworks such as Mimikatz. As with other critical vulnerabilities such as Zerologon, is it highly likely that this vulnerability will be leveraged by ransomware gangs in the near future. (Class, 2021)”

5 Implementation of the developed scanners

In this section, vulnerability scans created in Nessus and OpenVAS have been carried out on 2 university servers with applications under development. The type of tests carried out were white box tests, due to prior knowledge of the infrastructure. In this case we will only focus on vulnerabilities classified as high or critical, because these are the most serious vulnerabilities, and this is what the university's IT department is trying to correct. The Malware and credentialed patch scanners have not been executed, because to perform these scans the user and password of the server were required, which for security reasons have not been given. PrintNightmare scan was also discarded, because the servers do not have printers, so I do not see it as necessary. It should be noted that the vulnerabilities found will not be mentioned. This is because at the time of the delivery of this final degree project, the institution that has let me do the testing on their servers has not had time to fix them.

5.1 Nessus scans results

Below are the results obtained by the 8 Nessus scanners running on the servers.

5.1.1 Discovery Scan results

Discovery Scan has made a list of the open ports of the machines. In the case of the myapsit machine, only 2 ports are open: 22 and 3306. Knowing this information, we can assume that this machine will have a very small number of vulnerabilities. In the case of the machine appssids1t the following ports have been detected: 22, 80, 1091, 1092, 1093, 1094, 1095, 1096, 1097, 1098, 1099, 1100, 1101, 1102, 1102, 1004, 1105, 1106, 1107, 1108, 1109, 1110, 1111, 1112, 1113, 8080. The time it took to run the scanner was 17 minutes.

<input type="checkbox"/> Host ▼	FQDN	Ports	
<input type="checkbox"/> myapsit.local	myapsit.local	22, 3306	✖
<input type="checkbox"/> appssids1t.local	appssids1t.local	22, 80, 1091, 1092, 1093, 1094, 1095, 1096, 1097, 1098, 1099, 1100, 1101, 1102, 1103,...	✖

Figure 13 - Nessus Discovery scan results

5.1.2 Basic scan results

Thanks to Basic Scan a total of 22 vulnerabilities have been found on the appssids1t machine. Of which 1 is high, 20 are medium and 1 is low. In this case we will only focus on the high vulnerability.

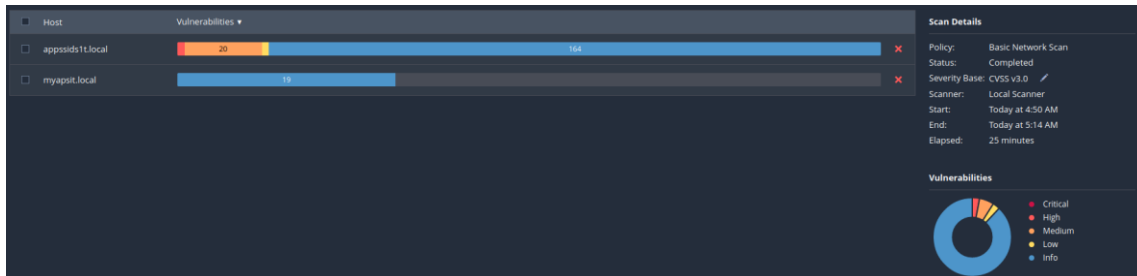


Figure 14 - Nessus basic scan results

It should be noted that Nessus ranks the most important vulnerabilities according to the Vulnerability Priority Rating (VPR) system. As we can see in the image, there are 3 vulnerabilities, all belonging to the appssids1t machine. The time it took to run the scanner was 25 minutes.

VPR Severity	Name	Reasons	VPR Score	Hosts
LOW	Apache Tomcat Remote Denial of Service Vulnerability	No recorded events	3.6	1
LOW	Apache Tomcat Remote Denial of Service Vulnerability	No recorded events	2.2	1
LOW	Apache Tomcat Remote Denial of Service Vulnerability	No recorded events	1.4	1

Figure 15 - Nessus basic scan VPR results

5.1.3 WEB scan results

This scanner detected a total of 77 vulnerabilities on the appssids1t machine, of which 47 were low, 28 medium, 1 high and 1 critical. On the myapsit machine no vulnerabilities were detected, which makes sense as it is a machine with a very small number of open ports. In this case we will only focus on the medium, high, and critical vulnerabilities.

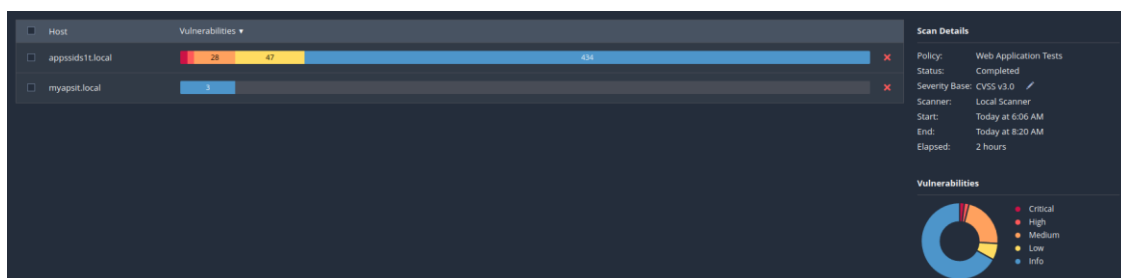


Figure 16 - Nessus Web scan results

In this scanner, as well as in the basic scan, we find the most important vulnerabilities as determined by VPR. Although 3 vulnerabilities are mentioned, note that there is only one new one, as the other 2 are the same ones that VPR detects in Basic scan. The time it took to run the scanner was 55 minutes.

VPR Severity	Name	Reasons	VPR Score ▼	Hosts
LOW	[REDACTED]	No recorded events	3.8	1
LOW	[REDACTED]	No recorded events	3.6	1
LOW	[REDACTED]	No recorded events	1.4	1

Figure 17 -Nessus web scan VPR results

5.1.4 MS Exchange scan results

In the case of MS Exchange scan, no vulnerability has been found on the 2 machines. The only thing is that the report tells us which ports are open and that we should protect them with an IP filter. This result is correct, as none of the machines use Exchange. The time it took to run the scanner was 9 minutes.

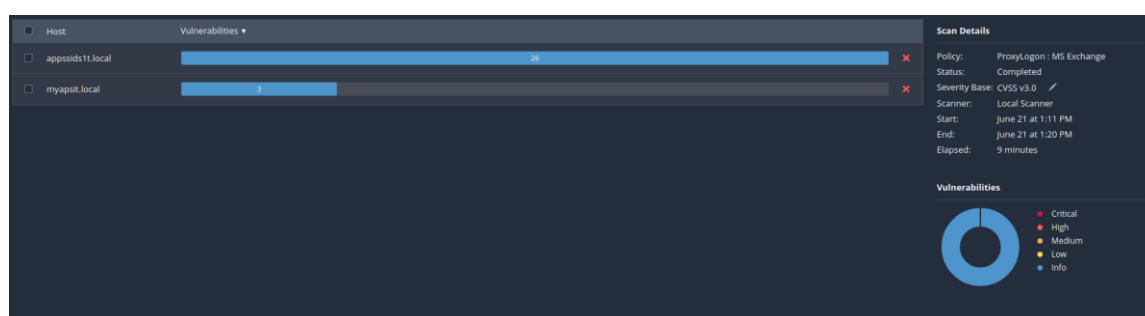


Figure 18 - Nessus Ms Exchange scan results

5.1.5 ContiLeaks scan results

With the ContiLeaks scan we have obtained the same result as with Ms Exchange scan, there is no vulnerability detected and we are recommended to set an IP filter to protect the open ports. The time it took to run the scanner was 8 minutes.

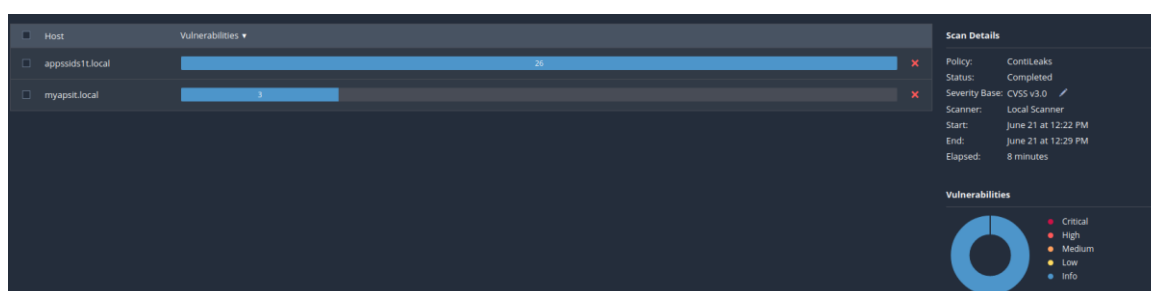


Figure 19 - ContiLeaks scan results

5.1.6 Log4Shell results

Log4Shell scan has not detected any vulnerability in the log4j service. Although it did not find any vulnerabilities, I think it is important to note that Nessus has listed the remote services, the plugins display for each host, determine if the hosts were alive by different ping types and identify the operating system. The time it took to run the scanner was 21 minutes.

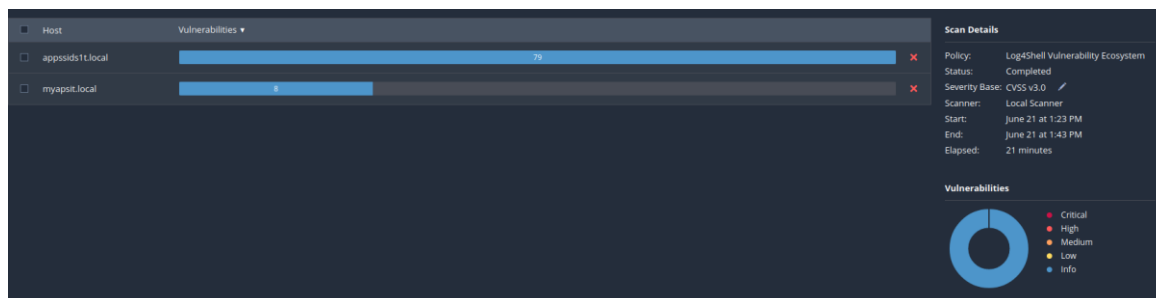


Figure 20 - Nessus log4shell scan results

5.1.7 Solorigate results

The Solorigate scanner has not found any vulnerabilities. However, it advises us to protect the appssids1t machine with an IP filter, to protect open TCP connections. The time it took to run the scanner was 11 minutes.

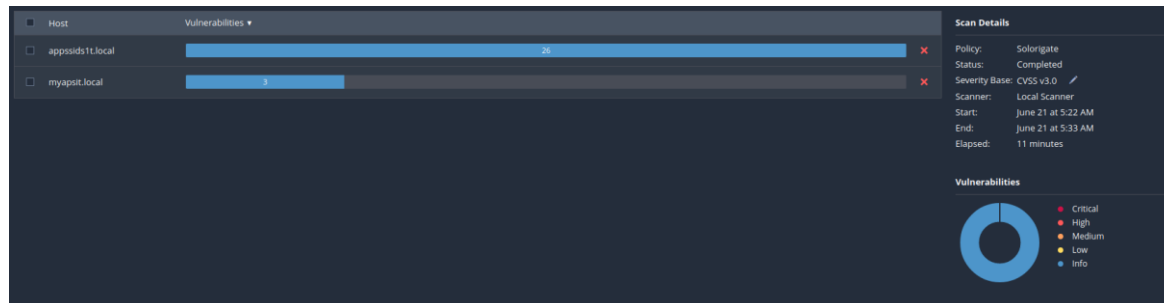


Figure 21 - Nessus solarigate scan results

5.1.8 Cisa scan results

The result of the Cisa scan is the same as the one presented by Solarigate, no vulnerabilities have been found and they warn to protect the appssids1t machine with an IP filter. The time it took to run the scanner was 15 minutes.

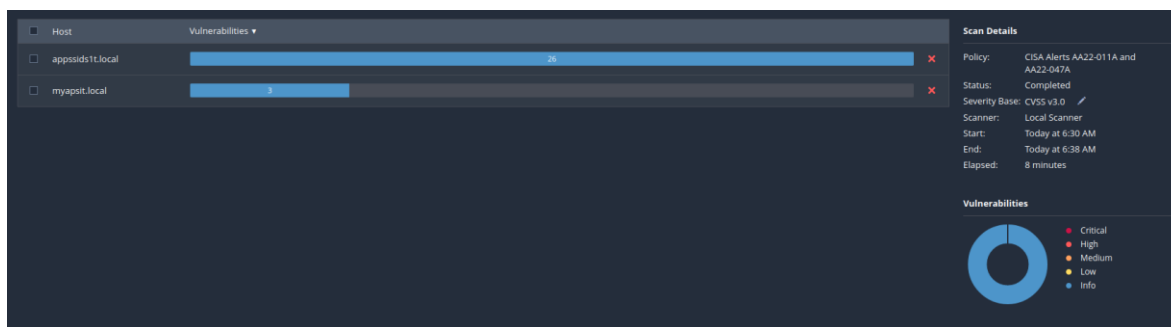


Figure 22 - Nessus cisa scan results

5.2 OpenVAS scans results

After running the Nessus scanners, I went on to run the OpenVAS scanners, highlighting that in OpenVAS a total of 4 scanners have been run.

5.2.1 Discovery scan results

The discovery scan took 1 hour and 19 minutes to complete. It detected a total of 27 ports, listed the 91 services found on the machines where the tests were performed and showed the operating system of the 2 servers, which is Debian.

Port	Hosts ▲
22/tcp	1
80/tcp	1
1091/tcp	1
1092/tcp	1
1093/tcp	1
1094/tcp	1
1095/tcp	1
1096/tcp	1
1097/tcp	1
1098/tcp	1
1099/tcp	1
1100/tcp	1
1101/tcp	1
1102/tcp	1
1103/tcp	1
1104/tcp	1
1105/tcp	1
1106/tcp	1
1107/tcp	1
1108/tcp	1
1110/tcp	1
1111/tcp	1
1112/tcp	1
1113/tcp	1
8080/tcp	1

Figure 23 - OpenVAS discovery scan port list

It has also made a list of CPE (Common Platform Enumeration) applications.





Application CPE ▼	
	cpe:/a:phpmyadmin:phpmyadmin:5.1.2
	cpe:/a:php:php
	cpe:/a:openbsd:openssh:8.4p1
	cpe:/a:jquery:jquery:3.6.0
	cpe:/a:jquery:jquery:1.7.1
	cpe:/a:john_beranek:meeting_room_booking_system:1.10.0
	cpe:/a:apache:tomcat:10.0.16
	cpe:/a:apache:http_server

Figure 24 - OpenVAS discovery scan CPE

5.2.2 Log4shell scan results

The Log4shell scan lasted a total of 33 minutes. It did not detect any log4j vulnerabilities.

Date	Status	Task
Thu, Jun 23, 2022 12:14 AM UTC	Done	appssids1t log4shell

Severity	High	Medium	Low	Log	False Pos.	Actions
0.0 (Log)	0	0	0	28	0	△ ×

Apply to page contents ▼ 🔍 ×

Figure 25 - OpenVAS log4shell scan results

5.2.3 Full and fast scan results

The Full and Fast scanner took 3 hours to complete and detected a total of 52 vulnerabilities, of which 51 are medium level and 1 is low level. It should be noted that it also listed the ports where these vulnerabilities are found, the vulnerabilities found in the operating system and the CPE vulnerabilities.

Date ▼	Status	Task
Wed, Jun 22, 2022 6:21 PM UTC	Done	appssids1t

Severity	High	Medium	Low	Log	False Pos.	Actions
6.8 (Medium)	0	51	1	231	0	△ ×

Apply to page contents ▼ 🔍 ×

Figure 26 - OpenVAS full and fast results

5.2.4 Full scan results

This scanner has found the same vulnerabilities as full and fast scan. This scanner took 4 hours to complete, as it performs a larger number of tests.

Date ▼

Thu, Jun 23, 2022 1:02 AM UTC

Status

Done

Task

apssi full scan

◀◀ 1 - 1 of 1 ▶▶

Severity

High

Medium

Low

Log

False Pos.

Actions

6.1 (Medium)

0

51

1

452

0

△ ×

Apply to page contents ▼

🔍 ✕

Figure 27 - OpenVAS full scan results

6 Conclusion

In this project, I have reached the objective that I set from the beginning, as I have created a vulnerability scanner system that allows us to improve the security of our servers. By having 2 programmes, such as OpenVAS and Nessus, this allows us to obtain more reliable results, as we are submitting the targets to be analysed in two different databases.

As for the creation of the scanners, from my point of view, Nessus has seemed more useful, as the configuration of the scanners has seemed simpler. While OpenVAS is more complex, as it offers many more possibilities for the configuration of scanners, which at the levels that we want to apply this system of scanners is not very useful. It should be noted that testing in OpenVAS through a VPN has been quite complicated, because if I activated the VPN directly in the virtual machine OpenVAS used the local network, not to mention that the time to perform the scans in OpenVAS has been much longer than the time of the Nessus scans.

As for the practical case where I have put into operation the scanners created in Nessus and OpenVAS. If we make the comparison Nessus has shown a higher performance, this is because it has identified a greater number of vulnerabilities and has done in less time. It should be noted that the department that owns the servers where the tests were carried out, also uses Nessus to carry out their own scans, although they only use the host scan and basic scan. Thanks to the application of the scans that I have created, specifically the WEB scan, 55 vulnerabilities have been detected, which the institution where they were carried out was not aware of. Of these 55 vulnerabilities, 1 was of critical level. Therefore, the importance of passing different vulnerability scans is demonstrated. It is worth noting that the scanners Log4Shell, Solorigate, ContiLeaks, MS Exchange and Cisa did not detect any vulnerabilities. This has an explanation,

as these scanners look for very specific vulnerabilities for certain services, as these servers belong to applications that are under development, some services have not yet been implemented or are not needed, such as Exchange, or when configuring the servers an attempt has been made to fix some vulnerabilities such as those that might belong to the log4j service.

7 Agreements

I would like to start by thanking my tutor, Pere Pons, for this final degree project. From the beginning of the project, he has been very present, he has organised weekly meetings, he has helped me with the doubts that have arisen throughout the work and, most importantly, he has helped me to get in touch with those responsible for cybersecurity at the Autonomous University of Barcelona. For me he is a clear example of what a tutor should be in this type of work. I would also like to thank Imma Gamo Nieto, head of the cybersecurity department, as she has allowed me to put into practice the scanners developed in the work. Finally, I would like to thank my family for their support throughout the development of this project. since being a field where I did not have much training, I have encountered frustrating situations, but they have always been there to help me overcome such situations.

8 References

Managing the Web Interface Access — Greenbone Enterprise Appliance 21.04.19

documentation. (n.d.). Docs.greenbone.net. Retrieved June 27, 2022, from

<https://docs.greenbone.net/GSM-Manual/gos-21.04/en/web-interface-access.html>

Scanning a System — Greenbone Enterprise Appliance 21.04.19 documentation. (n.d.).

Docs.greenbone.net. Retrieved June 27, 2022, from <https://docs.greenbone.net/GSM-Manual/gos-21.04/en/scanning.html>

Scanning a System — Greenbone Security Manager (GSM) 6 documentation. (n.d.).

Docs.greenbone.net. Retrieved June 3, 2022, from <https://docs.greenbone.net/GSM-Manual/gos-6/en/scanning.html?highlight=alive%20tests>

Reports and Vulnerability Management — Greenbone Enterprise Appliance 21.04.14

documentation. (n.d.). Docs.greenbone.net. <https://docs.greenbone.net/GSM-Manual/gos-21.04/en/reports.html>

Al Nuaimi, E., Al Neyadi, H., Mohamed, N., & Al-Jaroodi, J. (2015). Applications of big data to smart cities. *Journal of Internet Services and Applications*, 6(1).

<https://doi.org/10.1186/s13174-015-0041-5>

Alamer, M., & Almaiah, M. A. (2021, July 1). *Cybersecurity in Smart City: A Systematic Mapping Study*. IEEE Xplore. <https://doi.org/10.1109/ICIT52682.2021.9491123>

Elmaghraby, A. S., & Losavio, M. M. (2014). Cyber security challenges in Smart Cities: Safety, security and privacy. *Journal of Advanced Research*, 5(4), 491–497.

<https://doi.org/10.1016/j.jare.2014.02.006>

Epson Europe. (n.d.). *FAQ Article Page | Epson Europe*. Wwww.epson.eu. Retrieved June 27, 2022, from https://www.epson.eu/en_EU/faq/KA-01651/contents?loc=en-us#:~:text=The%20Print%20Spooler%20is%20software

Fong, S. L., Wui Yung Chin, D., Abbas, R. A., Jamal, A., & Ahmed, F. Y. H. (2019). Smart City Bus Application With QR Code: A Review. *2019 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS)*.

<https://doi.org/10.1109/i2cacis.2019.8825047>

Glass, G. (2021, July 8). *What is PrintNightmare? PrintNightmare Security Advisory*. Redscan. <https://www.redscan.com/news/printnightmare-security-advisory/>

Goud, N. (2021, January 7). *What is Solorigate*. Cybersecurity Insiders.

<https://www.cybersecurity-insiders.com/what-is-solorigate/>

Hafnium Hack: what happened? (2022, June 9). Safe-Connect. <https://safe-connect.com/the-hafnium->

[hack/#:~:text=The%20Hafnium%20Hack&text=Vulnerabilities%20in%20Microsoft%20Exchange%20servers](https://safe-connect.com/the-hafnium-hack/#:~:text=The%20Hafnium%20Hack&text=Vulnerabilities%20in%20Microsoft%20Exchange%20servers)

hammad, M. (2020, July 29). *Introduction of Novell NetWare*. GeeksforGeeks.

<https://www.geeksforgeeks.org/introduction-of-novell-netware/>

Infosec ISR Congress. (2014).

https://archive.nyu.edu/bitstream/2451/14999/2/Infosec_ISR_Congress.pdf

JasonGerend. (2018, July 9). *Overview of file sharing using the SMB 3 protocol in Windows*

Server. Microsoft.com. <https://docs.microsoft.com/en-us/windows-server/storage/file-server/file-server-smb-overview>

Kalinin, M., Krundyshev, V., & Zegzhda, P. (2021). Cybersecurity Risk Assessment in Smart

City Infrastructures. *Machines*, 9(4), 78. <https://doi.org/10.3390/machines9040078>

López, G. (2022, February 24). *Attack Graph Response to US-CERT AA22-011A & AA22-*

047A: Preparing for Russian State-Sponsored Cyberthreats. AttackIQ.

<https://attackiq.com/2022/02/24/preparing-for-russian-state-sponsored-cyberthreats-to-u-s-and-allied-critical-infrastructure/>

Mehta, Y., Manohara Pai, M. M., Mallissery, S., & Singh, S. (2016, March 1). *Cloud enabled*

air quality detection, analysis and prediction - A smart city application for smart

health. IEEE Xplore. <https://doi.org/10.1109/ICBDSC.2016.7460380>

Oladimeji, S., & Michael Kerner, S. (2021, June 16). *SolarWinds hack explained: Everything*

you need to know. WhatIs.com.

<https://www.techtarget.com/whatis/feature/SolarWinds-hack-explained-Everything-you-need-to-know#:~:text=SolarWinds%20is%20a%20major%20software>

Pitrelli, M. B. (2022, April 14). *Leaked documents show notorious ransomware group has an HR department, performance reviews and an “employee of the month.”* CNBC.

<https://www.cnbc.com/2022/04/14/conti-ransomware-leak-shows-group-operates-like-normal-tech-company.html#:~:text=Conti%20>

UCL. (2018, January 24). *What is SSH and how do I use it?* Information Services Division.

<https://www.ucl.ac.uk/isd/what-ssh-and-how-do-i-use-it>

UEStudio, & UESTudio. (2021, June 4). *El 60% de las empresas que sufren un ciberataque se ven obligadas a cerrar.* Ahora Más Cerca.

<https://ahoramascerca.elmundo.es/ciberseguridad/el-60-de-las-empresas-que-sufren-un-ciberataque-se-ven-obligadas-a-cerrar>

Watkins, B. (2014). *The Impact of Cyber Attacks on the Private Sector*

<http://www.amo.cz/wp-content/uploads/2015/11/amocz-BP-2014-3.pdf>

Welcome to Nessus 10.2.x (Nessus). (2019, May 1). Docs.tenable.com.

https://docs.tenable.com/nessus/10_2/Content/GettingStarted.htm