Project overview for portfolio upload

Project Title:

Network Penetration Test & Malware Analysis

Project Description:

Found on abstract on next page.

Tooling used (but not limited to):

- NMAP
- Nikto
- Enum4linux
- Nessus
- Hashcat
- Metasploit
- Any.run
- strings

Processes and Techniques used (but not limited to):

- Port Scanning
- Enumeration
- Vulnerability Scanning
- System Hacking (Rejetto exploit & Reverse Shell Exploit)
- Static & Dynamic Malware Analysis

Grade: A

Appendices have been removed to avoid plagiarism from future students, however I am happy to provide additional information or discuss the work further if desired.

Disclaimer*: I do not give permission for this work to be copied, those who do may be liable for plagiarism.*

Abstract

The following report details the process of a network penetration test and malware analysis. The aim of this penetration test and the accompanying report was to identify potential weaknesses & vulnerabilities within the network infrastructure of "Company X". To provide justification of highlighting these vulnerabilities, in this report, related exploits have been executed to show the severity of the found vulnerabilities. The process of penetration testing allows for a comprehensive report to be generated on both Company X's network architecture and its organisational context as well. Company X had previously found and identified malware on their system before putting the infected files in a sandboxed system. Therefore, an additional part of this report is to analyse the malware both statically and dynamically in the sandboxed environment to identify the contained malware and how the malware operates.

To carry out the penetration test and malware analysis, specific tools were required. A combination of a Windows 10 and a Kali Linux workstation was used to carry out the technical elements of this report. Industry standard 'Nmap' was used to scan open TCP & UDP ports, whilst tools such as enum4linux & nbtenum were used for enumeration to find out about Company X's network architecture, employee lists & contained policies. A Nessus vulnerability scan was performed to assess the severity of vulnerabilities and provide more context to the information obtained in the scanning & enumeration phases. System & password hacking was carried through exploiting a Rojetto exploit on a HTTP file server. XfreeRDP was used to gain remote access to the server as a domain admin because of this exploit. Another exploit was used in the form of a Reverse PHP shell to assist in manipulating file filtering on the company's server. Static malware analysis was carried out using Virus Total & strings, whilst 'App Any Run' allowed for the dynamic analysis to take place alongside executing the malware in the sandbox environment.

The key findings of this report determined that Company X had many weaknesses within their network that needed to be resolved as soon as possible. This weak security can be attributed to the malware being found on the system; this malware was found through analysis to be 'WannaCry' ransomware. Multiple exploits were able to be executed giving remote attackers avenues to access administrative areas on the network. Countermeasures are outlined at the end of the report, and it is suggested that Company X uses this information to update its network security policy and procedures.

Contents

1 Introduction........1
1.1 Background........1
1.2 Aims........2
2 Procedure........3
2.1 Overview of Procedure........3
2.2 Scanning........4
2.2.1 NMAP........4
2.2.2 TCP Port Scanning........4
2.2.3 UDP Port Scanning........7
2.2.4 Nikto Web Scanner........8
2.3 Enumeration........9
2.3.1 Identifying Domain Admins........9
2.3.2 Organisational Groups........10
2.3.3 Company Shares........10
2.3.4 Password Policy........11
2.4 Nessus Scan........13
2.4.1 Overview of results........13
2.4.2 Critical Vulnerabilities........14
2.4.3 High Vulnerabilities........15
2.4.4 Medium Vulnerabilities........16
System Hacking........17
2.4.5 Exploit 1 – 'Rejetto'........17
2.4.6 Exploit 2 – 'PHP Reverse Shell'........19
2.5 Malware Analysis........20
2.5.1 Static Analysis........20
2.5.2 Dynamic Analysis........22
3 Discussion........25
3.1 General Discussion........25
3.2 Countermeasures........26

3.2.1 Training & Education 26
3.2.2 Upgrading Legacy Software 26
3.2.3 Removing Redundancy on servers 26
3.2.4 Password Policy 27
3.2.5 Privilege Management 27
3.2.6 Firewalls 27
3.3 Future Work 27
References 28
Appendices .... **Error! Bookmark not defined.**
Appendix A – Port Scanning .... **Error! Bookmark not defined.**
Appendix B – Enumeration .... **Error! Bookmark not defined.**
Appendix C – Nessus Vulnerability Scans .... **Error! Bookmark not defined.**
Appendix D – System Hacking .... **Error! Bookmark not defined.**
Appendix E – Malware Analysis .... **Error! Bookmark not defined.**
Appendix F – HTTP Servers .... **Error! Bookmark not defined.**

1 INTRODUCTION

1.1 BACKGROUND

The pervasiveness of online networks within society has vastly changed the way in which society operates both on and offline. Many individuals make conscious efforts to conceal their life away from online avenues as much as possible whereas others are quick to enter their personal and private details to a multitude of online technologies.

As is the way with new technology, society finds reasons to implement them in every aspect of modern life, whether it be via work, personal or private life. Local businesses to multi global organisations are all reliant on the functionality delivered by technology. If this functionality were to be interfered with, it could create a great deal of damage for the businesses that are reliant on the affected technology. In fact, according to the international Journal of Cyber Warfare and terrorism, data breach costs in the USA rose from 3.86 million USD to 4.24 million in 2021 (Sebastian, 2022).

Therefore, companies who have their own networking systems are ensuring it is completely secured from outside threats. This shift towards companies being more security focused is essential to develop a workforce that are highly knowledgeable on security measures and thus minimising threat vectors online (Alhayani *et al.,* 2021). However, ensuring every employee within a company is up to date on the newest security principles is a very difficult job, therefore vulnerabilities still exist. In this, there beckons the problem, how can those developing new technology have security at the forefront of their mind. How can IT professionals ensure that their company is up to date with the correct security ?

This is where penetration testing can deliver solutions to companies for this very problem. Penetration testing can help a company identify the most critical vulnerabilities and suggest changes based on the required level of remediation (Naik *et al.,* 2009). Through a comprehensive penetration test, employees, administrators and company executives will all be educated on the best security practices.

1.2 AIMS

The primary aim of this report is to present “Company X” with a comprehensive report on their current network security. This report will be created through documenting the process of a network penetration test. The penetration test will simulate the processes a remote attacker would take when trying to breach Company X’s network security. By carrying out this process, it will allow for vulnerabilities to come to light, and exploits related to these vulnerabilities to be tested. All information obtained through the penetration test will be used to provide context to the technical processes used. Countermeasures will be developed at the end of the report to show how company X’ – if required – can implement stronger network security and a greater security ethos at their company. To recap, the aims of this report are the following.

- Develop a comprehensive report on the vulnerabilities that exist on company X’s network.
- Discover valid exploits and determine the extent to which Company X is exposed to the vulnerabilities discovered during the scanning phases.
- Include discussion on these results and…
    - Develop a set of countermeasures to the vulnerabilities and exploits discovered through the penetration test,
        - Enable company X to use these countermeasures to improve their security going forward.
- Analyse the malware that is sandboxed on their system, identify how this malware operates, what the malware is & how it got onto the network.

2 Procedure

2.1 Overview of Procedure

The penetration test of Company X's network was done by following a systematic approach. This systematic approach is split into the following parts: Network Scanning; Enumeration; Vulnerability Scanning and finally System Hacking. The malware was then analysed both statically and dynamically using various methods that are outlined in a further section.

When carrying out the penetration testing process, a combination of Kali Linux & a Windows 10 VM was used as the primary workstations. This allowed for comprehensive testing, ensuring a variety of tools were used to cross reference results for accuracy. Below is an outline of the methods used to obtain the results. The following subsections will break down into the following.

- Scanning of 192.168.10.1 (Server 1) & 192.168.10.2 (Server 2)
  - NMAP - TCP & UDP
  - Nikto - Run on Identified ports of interest to provide further information.
- Enumeration.
  - Enumeration using ***Enum4Linux*** on Kali Linux & ***NBTE*** on windows.
  - (Additional results from these individual scans can be found in Appendix 01:
    - (*NBT Scan, RPCclient, Crackmapexec, SMTP, SNMP*).
- Vulnerability Scanning
  - Nessus
- System Hacking
  - Password with Hash Cat & dictionaries.
  - Metasploit
  - PHP Reverse Shell through bypassing filters
  - Remote Command Execution using Rojetta exploit.
    - Xfreerdp to gain remote access to the server.
- Malware Analysis
  - Virus Total
  - Strings
  - App Any Run

2.2 Scanning

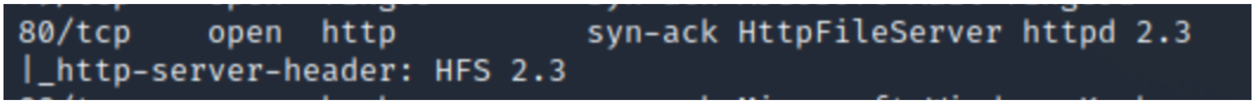
2.2.1 NMAP

NMAP was utilised to systematically explore the network architecture, identifying TCP and UDP open ports on server 1 and server 2 at IP addresses 192.168.10.1 and 192.168.10.2, respectively. The NMAP scan provided insights to the associated services & their versions on the open ports. Collecting this information is essential for the remainder of the penetration test as it allows for research to be carried out into the services on these ports to determine if there are known vulnerabilities and thus allow for subsequent testing & potential exploitation.

2.2.2 TCP Port Scanning

2.2.2.1 192.168.10.1

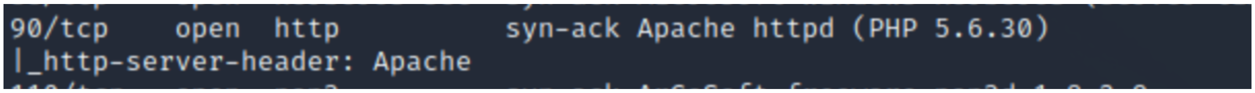
Server 1 was scanned for all open TCP ports. This provided some interesting results as some of the information here proved to be the foundation for an exploit further into the testing process. Such as Port 80 which is hosting an instance of Http File Server 2.3. This was important as some additional research into potential vulnerabilities surrounding http file servers with version 2.3 quickly returned results pointing to an exploit on ExploitDB known as the 'Rejetto HTTP File Server – Remote Code Execution exploit' (Thapa, 2016). This exploit is covered further in the system hacking subsection of this report and provides some guidance on how vulnerable Company X is on this server.



```
80/tcp    open  http          syn-ack HttpFileServer httpd 2.3
|_http-server-header: HFS 2.3
```

[Figure 1: NMAP Scan of TCP ports on 192.168.10.1]

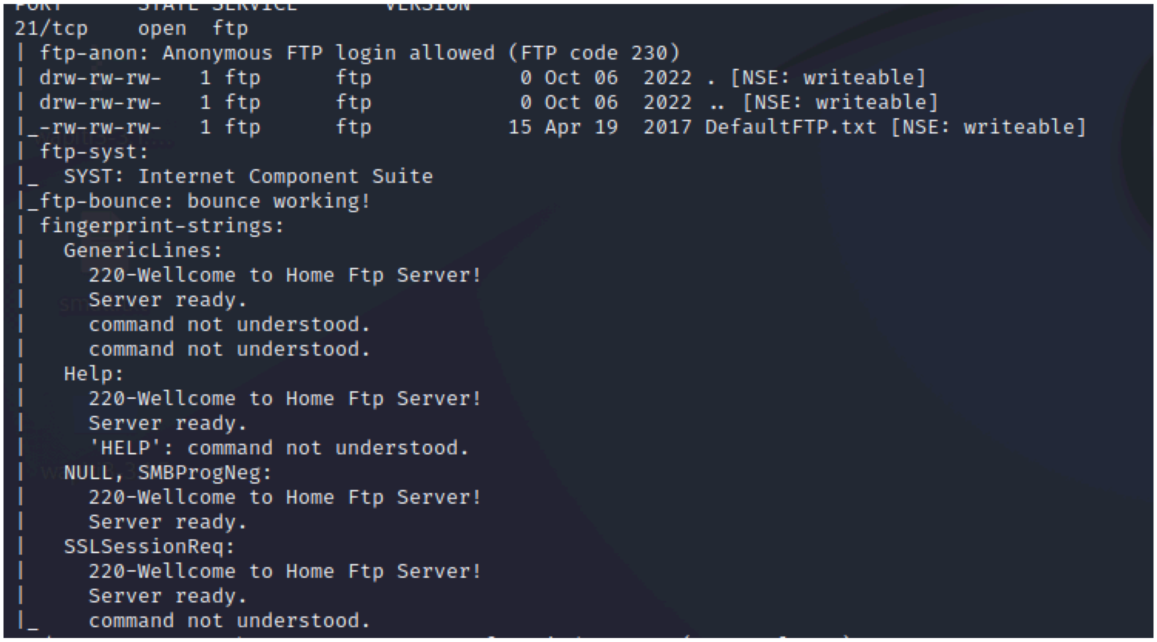
Following on from this, there is also another port running HTTP as a service, port 90. Port 90 is an Apache HTTP Server running PHP 5.6.30. When looking at the Nessus scans at a later stage of the testing (highlighted in the vulnerability scanning section) Nessus has identified some critical vulnerabilities surrounding servers running this PHP version. Like the fileserver example before, this has also produced an exploit in the system hacking section of this report.



```
90/tcp    open  http          syn-ack Apache httpd (PHP 5.6.30)
|_http-server-header: Apache
```

[Figure 2: Nmap Scan of Apache server on port 90]

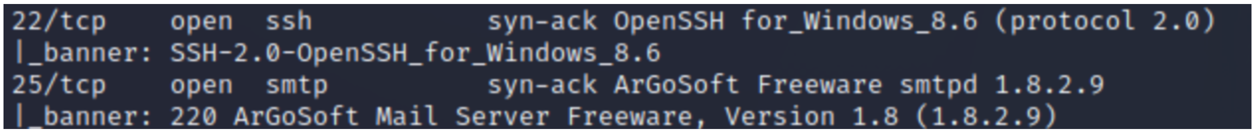
There was additional information that was also gained from the TCP scan on server 1, whilst they did not lead to further exploits, it is important to note that these are vulnerabilities. For example, running on open port 21 is an FTP file server that it allows for anonymous login. Anonymous login is dangerous for the security on Company X's network as it can allow for anyone to login without providing any authentication credentials. This means that unauthorised individuals can potentially gain access to this file server.



```
PORT      STATE SERVICE       VERSION
21/tcp    open  ftp
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
| drw-rw-rw-   1 ftp      ftp             0 Oct 06  2022 . [NSE: writeable]
| drw-rw-rw-   1 ftp      ftp             0 Oct 06  2022 .. [NSE: writeable]
|_-rw-rw-rw-   1 ftp      ftp            15 Apr 19  2017 DefaultFTP.txt [NSE: writeable]
| ftp-syst:
|_  SYST: Internet Component Suite
|_ftp-bounce: bounce working!
| fingerprint-strings:
|   GenericLines:
|     220-Wellcome to Home Ftp Server!
|     Server ready.
|     command not understood.
|     command not understood.
|   Help:
|     220-Wellcome to Home Ftp Server!
|     Server ready.
|     'HELP': command not understood.
|   NULL, SMBProgNeg:
|     220-Wellcome to Home Ftp Server!
|     Server ready.
|   SSLSessionReq:
|     220-Wellcome to Home Ftp Server!
|     Server ready.
|_    command not understood.
```

[Figure 3: FTP Anonymous login]

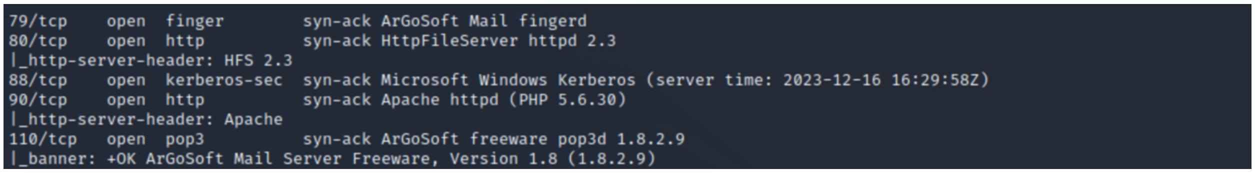
Analysing the TCP scans further, there were several other ports that were all running a variety of different services. Firstly, port 22 was running SSH for windows 8.6. Secondly, port 25 was running Argosoft Mail Server. As a mail server this can potentially be targeted by individuals with malicious ideals and is recommended that all email correspondence is correctly secured. ArGoSoft mail servers have been linked to remote attacks, such as arbitrary web scripts (CVE-2006-0978)



```
22/tcp    open  ssh           syn-ack OpenSSH for_Windows_8.6 (protocol 2.0)
|_banner: SSH-2.0-OpenSSH_for_Windows_8.6
25/tcp    open  smtp          syn-ack ArGoSoft Freeware smtpd 1.8.2.9
|_banner: 220 ArGoSoft Mail Server Freeware, Version 1.8 (1.8.2.9)
```

[Figure 4: ArGoSoft Mail servers]

Port 110 was running another ArGoSoft freeware, this time using pop3d. This was potentially a security concern as pop3d services are susceptible to brute force attacks. These attacks can lead to unauthorised access. Regular monitoring and efficient patching are required to reduce the risk faced from this server.

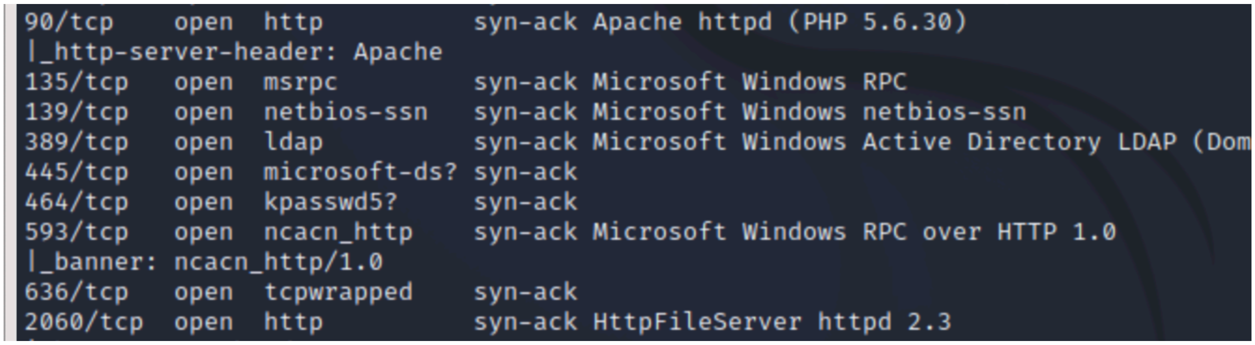


```
79/tcp    open  finger        syn-ack ArGoSoft Mail fingerd
80/tcp    open  http          syn-ack HttpFileServer httpd 2.3
|_http-server-header: HFS 2.3
88/tcp    open  kerberos-sec  syn-ack Microsoft Windows Kerberos (server time: 2023-12-16 16:29:58Z)
90/tcp    open  http          syn-ack Apache httpd (PHP 5.6.30)
|_http-server-header: Apache
110/tcp   open  pop3          syn-ack ArGoSoft freeware pop3d 1.8.2.9
|_banner: +OK ArGoSoft Mail Server Freeware, Version 1.8 (1.8.2.9)
```

[Figure 5: Pop3 Servers]

2.2.2.2 192.168.10.2

One critical piece of information gained from running a TCP scan on server two was that the same services that were found on server 1 can be seen running on server two. In this case, port 90 was running an Apache server with PHP 5.6.3. This means that server 2 was also open to being exploited in the same way as server 1. Once a remote attacker has access to one server it highly likely they could replicate their steps to do the same to the other server. The steps the attacker can go through are outlined in system hacking section of this report and they should show the danger of having the same services running on both servers.



```
90/tcp    open  http          syn-ack Apache httpd (PHP 5.6.30)
|_http-server-header: Apache
135/tcp   open  msrpc         syn-ack Microsoft Windows RPC
139/tcp   open  netbios-ssn   syn-ack Microsoft Windows netbios-ssn
389/tcp   open  ldap          syn-ack Microsoft Windows Active Directory LDAP (Dom
445/tcp   open  microsoft-ds? syn-ack
464/tcp   open  kpasswd5?     syn-ack
593/tcp   open  ncacn_http    syn-ack Microsoft Windows RPC over HTTP 1.0
|_banner: ncacn_http/1.0
636/tcp   open  tcpwrapped    syn-ack
2060/tcp  open  http          syn-ack HttpFileServer httpd 2.3
```

[Figure 6: Showing identical services on 192.168.10.1]

This point is echoed when looking at port 2060 which was also running a HTTP Fileserver 2.3. Port 53 was open with a Simple DNS Plus on server two. It is essential that this remains fully secured as any lax security on this port can enable the server to be open to a DDoS attack.

Following on, port 389 indicated there was a Prescence of a Windows Active Directory. This meant that it was likely the servers centralised directory service. This, combined with the critical vulnerabilities of port 90 and 2060 means that if someone were to gain access to the server, they would be able to access the windows directory. Once having accessed this port, the online attacker would be able to create administrators, upload files & hide malicious activity on the server with admin rights (Motero *et al.,* 2021).

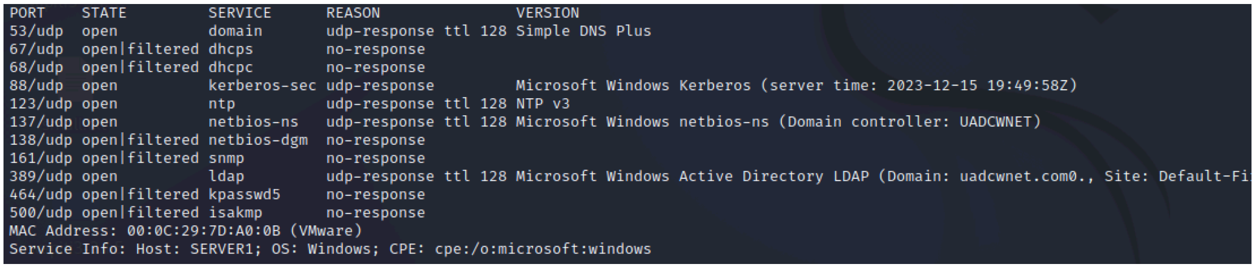
2.2.3 UDP Port Scanning

During UDP scanning there was several ports that seemed to indicate an open / filtered status. Whilst nothing inherently significant could be learnt from these ports it is important that they are recorded to provide a full overview of the network. Full UDP scans can be found in appendix.

Port 53 identified as open and is a simple DNS plus server. DNS servers are crucial in networks as they help the network function as normal, translating domain names to IP addresses. Therefore, any misconfiguration that could lead to a vulnerability could be critical.

Port 67 & 68 were showing open/filtered status. These ports are associated with DCHP. These ports do not have a direct response due to the nature of an open/filtered port therefore no further analysis is carried out with these ports. This is the same for ports 138, 161, 464 & 500. However, it is worth nothing that port 138 had a NetBIOS server running on it and this could have potentially been a security risk if misconfigured. Port 161 & 500 were running SNMP & ISAKMP respectively. These are potential avenues of attack for malicious actors due to SNMP being open to exploitation by unauthorised access (Jiang, 2002).

Port 389 had Microsoft Windows Active Directory on it, this is critical for directory services, as mentioned previously it can lead to malicious actors gaining access to the windows directory and potentially allowing themselves to gain admin privileges on the server, therefore it is important this is correctly secured.

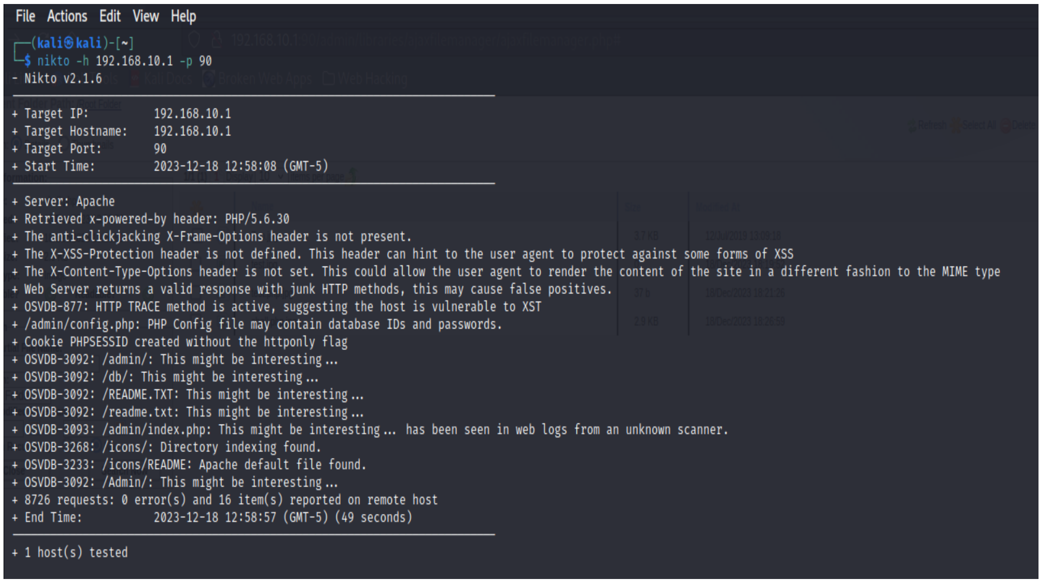


```
PORT    STATE         SERVICE      REASON               VERSION
53/udp  open          domain       udp-response ttl 128 Simple DNS Plus
67/udp  open|filtered dhcps        no-response
68/udp  open|filtered dhcpc        no-response
88/udp  open          kerberos-sec udp-response         Microsoft Windows Kerberos (server time: 2023-12-15 19:49:58Z)
123/udp open          ntp          udp-response ttl 128 NTP v3
137/udp open          netbios-ns   udp-response ttl 128 Microsoft Windows netbios-ns (Domain controller: UADCWNET)
138/udp open|filtered netbios-dgm  no-response
161/udp open|filtered snmp         no-response
389/udp open          ldap         udp-response ttl 128 Microsoft Windows Active Directory LDAP (Domain: uadcwnet.com0., Site: Default-Fi
464/udp open|filtered kpasswd5     no-response
500/udp open|filtered isakmp       no-response
MAC Address: 00:0C:29:7D:A0:0B (VMware)
Service Info: Host: SERVER1; OS: Windows; CPE: cpe:/o:microsoft:windows
```

[Figure 7: UDP Scans on 192.168.10.1]

2.2.4 Nikto Web Scanner

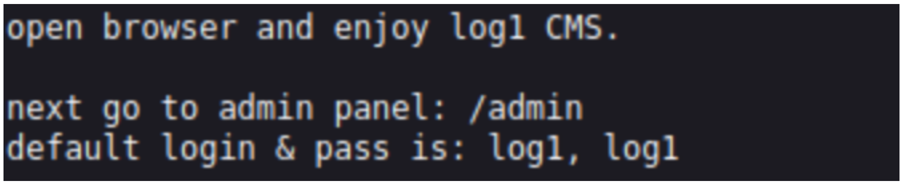
Once the initial Nmap scans were performed, further investigation was carried out into some of the more interesting web services discovered on open ports. These were ports 80 and 90 on server 1, ports 90 and 2060 on server two.



```
┌──(kali㉿kali)-[~]
└─$ nikto -h 192.168.10.1 -p 90
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.10.1
+ Target Hostname:    192.168.10.1
+ Target Port:        90
+ Start Time:         2023-12-18 12:58:08 (GMT-5)
---------------------------------------------------------------------------
+ Server: Apache
+ Retrieved x-powered-by header: PHP/5.6.30
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ Web Server returns a valid response with junk HTTP methods, this may cause false positives.
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
+ /admin/config.php: PHP Config file may contain database IDs and passwords.
+ Cookie PHPSESSID created without the httponly flag
+ OSVDB-3092: /admin/: This might be interesting...
+ OSVDB-3092: /db/: This might be interesting...
+ OSVDB-3092: /README.TXT: This might be interesting...
+ OSVDB-3092: /readme.txt: This might be interesting...
+ OSVDB-3093: /admin/index.php: This might be interesting... has been seen in web logs from an unknown scanner.
+ OSVDB-3268: /icons/: Directory indexing found.
+ OSVDB-3233: /icons/README: Apache default file found.
+ OSVDB-3092: /Admin/: This might be interesting...
+ 8726 requests: 0 error(s) and 16 item(s) reported on remote host
+ End Time:           2023-12-18 12:58:57 (GMT-5) (49 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

[Figure 8: Nikto Scan on 192.168.10.1:90]

The Apache server running on port 90, after it was scanned with Nikto, revealed hidden pages on the HTTP server that were not previously there. On these webpages, a hidden text file titled ‘readme.txt’ was discovered. Upon closer inspection, this text file was related to the server’s configuration. The information obtained on this text file allowed certain pages to be accessed that were responsible for some of the exploits. They later allowed login details for further pages; this process is covered more in depth within the system hacking section, however the relevant contents of the readme can be seen in figure below.



```
open browser and enjoy log1 CMS.

next go to admin panel: /admin
default login & pass is: log1, log1
```

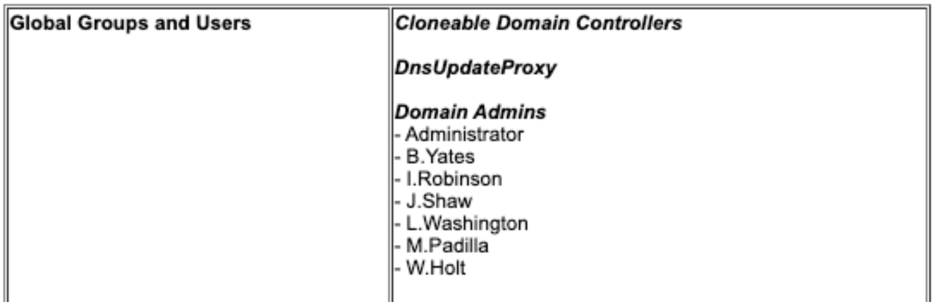
[Figure 9: Log1CMS admin login]

2.3 Enumeration

There are many tools that could have been used for enumeration, however after consideration, a combination of Enum4linux on Kali Linux and nbtenum (NetBIOS Enumerator) on Windows were decided upon. This was for numerous reasons, one of which was the readability of each of the reports. Both enum4linux and nbtenum have a direct output function which prints all information obtained by the tools into a .txt and .html file respectively. This allows important information to be obtained, stored and utilised both in an efficient and coherent manner. Other tools were used to carry out enumeration and these outputs can be seen in section B of the appendices.

2.3.1 Identifying Domain Admins

When looking at the output given by nbtenum, it was easily identifiable who the domain admins were. The domain admins were the same for both servers. Therefore, the below figure is valid for both server 1 & 2. Understanding who the domain admins are is important for numerous reasons. Privilege escalation opportunities is a vulnerability whereby an individual can identify which users belong to groups that have administrative rights or access to sensitive information on the company's network and target them for complete access of the server.

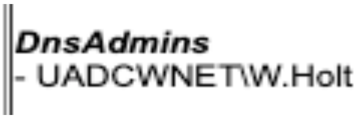


Global Groups and Users	***Cloneable Domain Controllers*** ***DnsUpdateProxy*** ***Domain Admins*** - Administrator - B.Yates - I.Robinson - J.Shaw - L.Washington - M.Padilla - W.Holt

[Figure 10: Domain Admins]

2.3.1.1 DNS Admin

A piece of information worth taking into consideration for the system hacking section is the DNS admin listed below & the previously obtained information of multiple open ports with DNS existing.

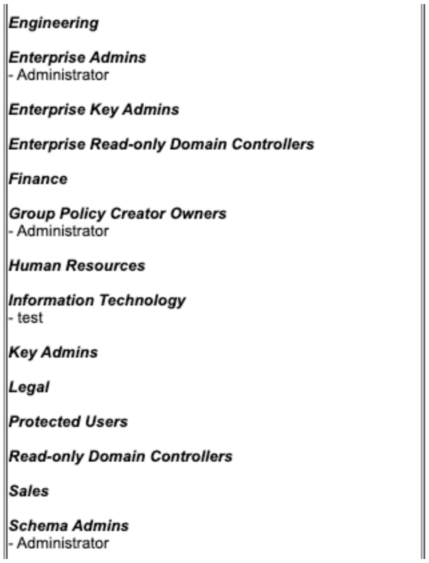


DnsAdmins
- UADCWNET\W.Holt

[Figure 11: DNSAdmin]

2.3.2 Organisational Groups

When analysing the results, more details about the inner workings of the company can be learnt, such as the groups that have been created on the system. This allows some guided assumptions to be made about the organisational structure of the company. For example, identifying the 'Finance' group, further questions can arise, such as, can any financial information be obtained, or any manipulation of financial records be carried out. Similarly, when looking at the core function of a business, is it possible for an individual to gain access to information about this core function and exploit this, perhaps affect the daily running of a business, think of the groups Engineering & Sales found in the nbteunm scan of server 1 in Figure 12 below.

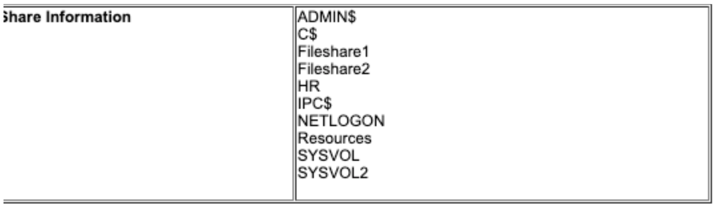


```
Engineering

Enterprise Admins
- Administrator

Enterprise Key Admins

Enterprise Read-only Domain Controllers

Finance

Group Policy Creator Owners
- Administrator

Human Resources

Information Technology
- test

Key Admins

Legal

Protected Users

Read-only Domain Controllers

Sales

Schema Admins
- Administrator
```

[Figure 12: Organisational Groups]

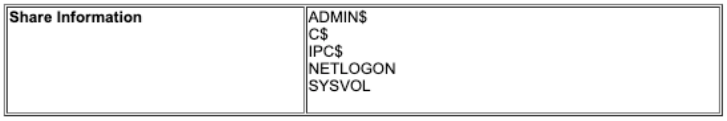
2.3.3 Company Shares

Following on from identifying the company structure, learning how the company has their shares set up will help provide more information on what the company relies on to operate and what functions of the business is stored on the servers. This can be identified through looking at the shares below on both server 1 and server 2. By identifying which shares are hidden (indicated by $) there can be some assumptions made on the shares that could be important to the company and therefore worth hiding.



Share Information	ADMIN$ C$ Fileshare1 Fileshare2 HR IPC$ NETLOGON Resources SYSVOL SYSVOL2

[Figure 13: Share Information from 192.168.10.1]

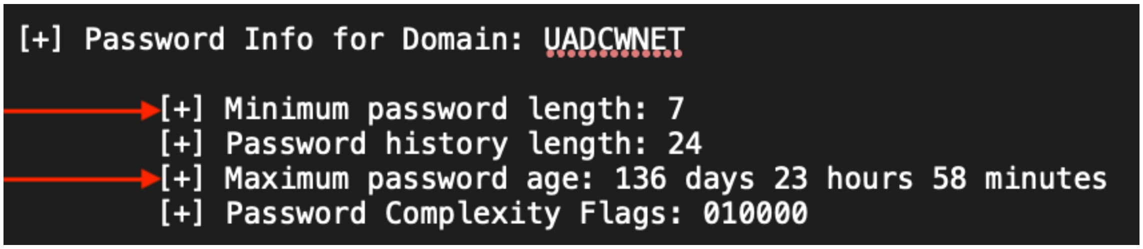


Share Information	ADMIN$ C$ IPC$ NETLOGON SYSVOL

[Figure 14: Share information from 192.168.10.2]

2.3.4 Password Policy

There are some key takeaways from the password policy obtained by Enum4Linux. Firstly, the password minimum length is set to 7 characters, less than the recommended amount by NIST (Weir *et al.,* 2010), whilst this is not much lower, it is still important to adhere to strict security practices where possible, particularly when recommended by a large standards agency such as NIST. This is like the maximum password age, which is set at 136 days, whilst this is close to the recommended max password age value that is determined by windows (Windows, 2022) it still doesn't meet the requirements entirely. These near misses are reflective a security policy that is limited in effectiveness.

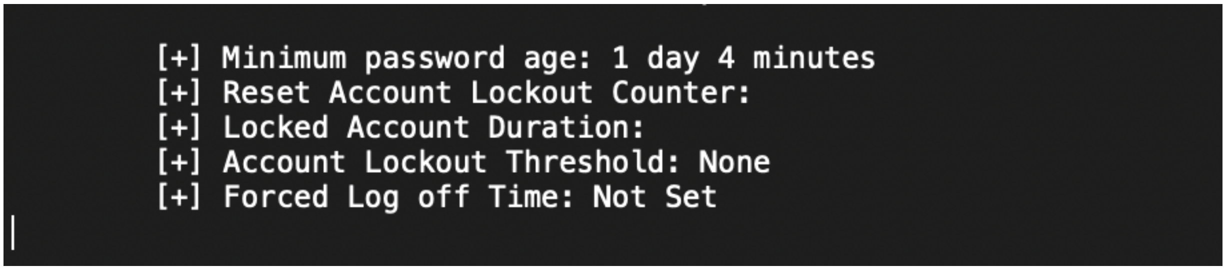


```
[+] Password Info for Domain: UADCWNET

        [+] Minimum password length: 7
        [+] Password history length: 24
        [+] Maximum password age: 136 days 23 hours 58 minutes
        [+] Password Complexity Flags: 010000
```

[Figure 15: Password policy obtained from Enum4linux]

Password complexity flags are disabled, allowing for passwords that contain all the same character, repeating numbers and any common words or expressions can all be used as part of the passwords created on the network. No password complexity requirements & minimum required length of a password ensures that system will be extremely vulnerable to both dictionary & brute force password attacks. This is performed in the system hacking section to reflect how simple it can be for a malicious actor to obtain passwords on the network.

The vulnerabilities generated through a lacklustre password policy are amplified further when observing that there was no lockout policy. Lockout threshold can determine the number of failed sign in attempts that will cause the user trying to log in to be locked out of the account. In having the threshold set to none, this will allow automated attacks such as brute force attacks to be used without the malicious actor having to mitigate their attempts because of fear of being logged out.



```
        [+] Minimum password age: 1 day 4 minutes
        [+] Reset Account Lockout Counter:
        [+] Locked Account Duration:
        [+] Account Lockout Threshold: None
        [+] Forced Log off Time: Not Set
```

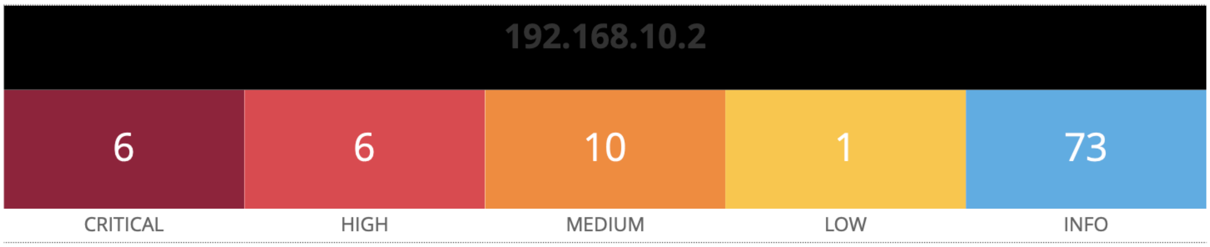
[Figure 16: Password policy continued]

Whilst this is less critical for remote attacks, no forced log off time is an issue. Not having a forced log off time can allow for a physical vulnerability to be created. It is very common for users to forget to log off machines after work or lock machines when away from their desk. Because of this, it can allow for any individual to gain physical access to the computer. Forced log off time prevents this to a certain degree as it will log users out after a period, ensuring systems can’t be accessed overnight or on weekends for example.

2.4 Nessus Scan

2.4.1 Overview of results

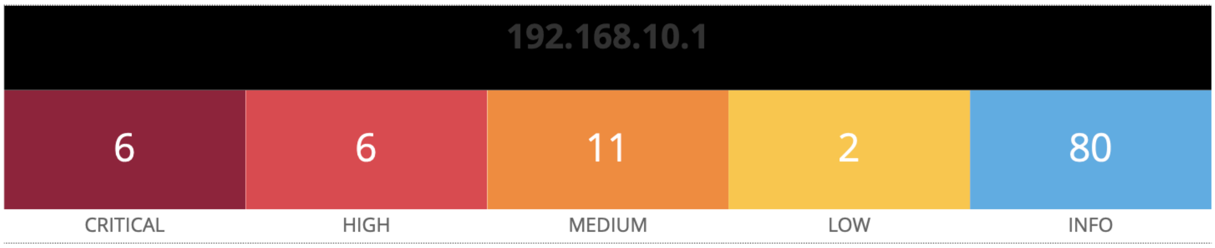
The Nessus vulnerability scan returned 96 results for server 1 and 105 results for server 2. When considering these results for both servers, it's important to note how Nessus categorises each type of vulnerability. For example, critical results from Nessus are beneficial for indicating where the company's biggest vulnerabilities are within its network. These critical results are typically centred around essential services to companies such as file servers, that, if exploited, can cause business outages or large data breaches. The results generated helped guide the focus of the penetration test when going into the system hacking phase. Typically, once the critical scans picked up by Nessus are exploited, the fallout from the exploit being activated on the server will lead to a variety of catastrophic breaches.



192.168.10.2

6	6	10	1	73
CRITICAL	HIGH	MEDIUM	LOW	INFO

[Figure 17: Nessus Results for server 192.168.10.1]

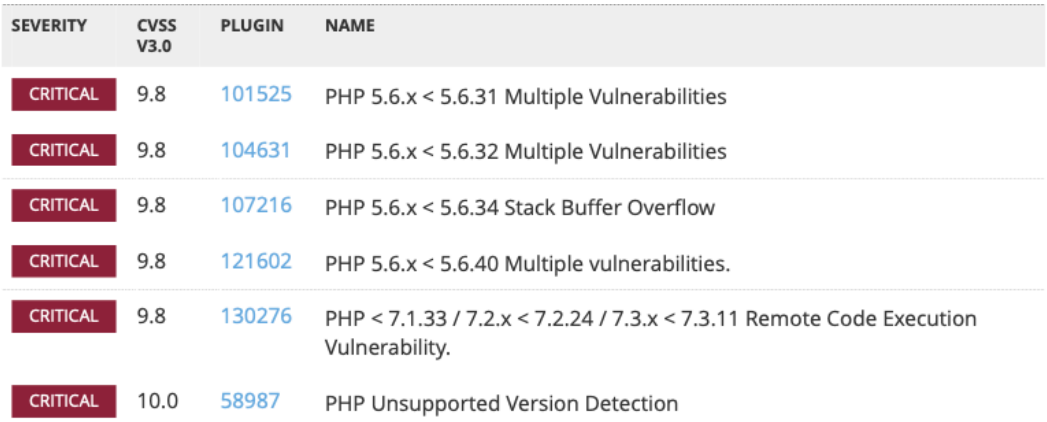


192.168.10.1

6	6	11	2	80
CRITICAL	HIGH	MEDIUM	LOW	INFO

[Figure 18: Nessus Results for server 192.168.10.2]

2.4.2 Critical Vulnerabilities



SEVERITY	CVSS V3.0	PLUGIN	NAME
CRITICAL	9.8	101525	PHP 5.6.x < 5.6.31 Multiple Vulnerabilities
CRITICAL	9.8	104631	PHP 5.6.x < 5.6.32 Multiple Vulnerabilities
CRITICAL	9.8	107216	PHP 5.6.x < 5.6.34 Stack Buffer Overflow
CRITICAL	9.8	121602	PHP 5.6.x < 5.6.40 Multiple vulnerabilities.
CRITICAL	9.8	130276	PHP < 7.1.33 / 7.2.x < 7.2.24 / 7.3.x < 7.3.11 Remote Code Execution Vulnerability.
CRITICAL	10.0	58987	PHP Unsupported Version Detection

[Figure 19: Critical vulnerabilities]

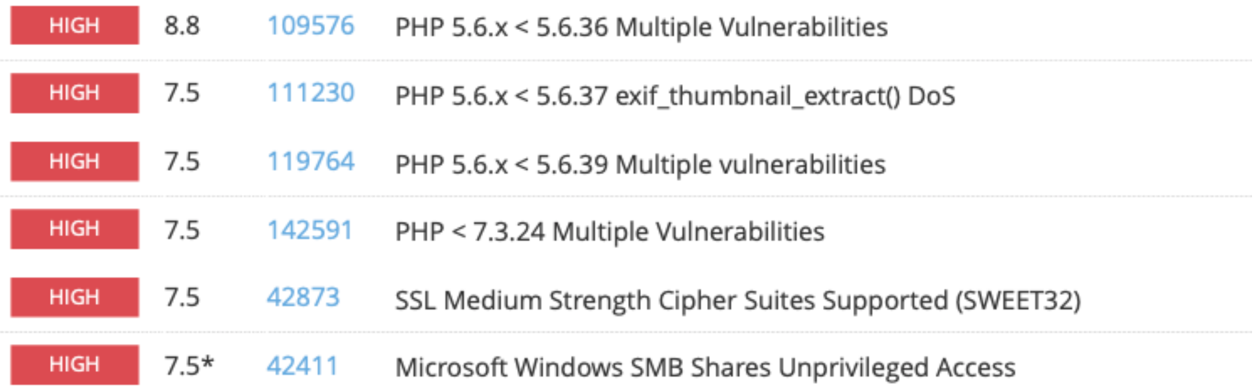
Figure 19 reported the critical vulnerabilities found in both server 1 and server 2. Therefore, the damage that can be caused if these vulnerabilities were to be exploited could easily be doubled when there is the same service running on both servers. This duplication of services across servers is again showing why critical vulnerabilities are dangerous if they exist.

When analysing the results, the most pressing vulnerabilities on the server were related to PHP in some way. When further observing the '*PHP 5.6x < 5.6.36 multiple vulnerabilities*' on the Nessus database, more information is discovered regarding the encompassed variations of this vulnerabilities that can fall under the heading. Firstly, multiple out of bounds read errors existed in multiple locations on the server. As referenced on the Nessus database this can lead to remote attackers carrying out multiple malicious activities such as crashing a process link the PHP library, resulting in denial of service; disclosing sensitive memory contents; produce denial of service conditions & exhaust CPU resources (CVE, 109576) . All these exploits could be costly to Company X in both time and money. These types of vulnerabilities, when exploited, have the potential to stop business functions to a halt & risk data breach for sensitive crucial information that is held by Company X .

The PHP Unsupported Version Detection vulnerability received a score of 10 from Nessus. A rating of 10 Is allocated to a vulnerability when it has the potential to have catastrophic impact on the target when exploited. In this case, PHP unsupported version detection vulnerability is when a version of PHP that is running on the service Is unsupported. This is very dangerous for Company X to be running this legacy software on their servers where there is such an emphasis on storing information both on the employees and the company on these servers (Tervoort *et al.,* 2020), details which were obtained through previous enumeration techniques .

The next vulnerability that was looked at was the PHP < 7.1.33 / 7.2.x < 7.2.24 / 7.3.x < 7.3.11 Remote Code Execution Vulnerability. Looking at Remote Code Execution (RCE) as an exploit, it is particularly dangerous as hackers will be able to execute code on the breached server. This type of exploit is utilised within the system hacking section however, this is on a HTTP Fileserver, the type of malicious activities that can be carried out from an RCE is similar across the context.

2.4.3 High Vulnerabilities



HIGH	8.8	109576	PHP 5.6.x < 5.6.36 Multiple Vulnerabilities
HIGH	7.5	111230	PHP 5.6.x < 5.6.37 exif_thumbnail_extract() DoS
HIGH	7.5	119764	PHP 5.6.x < 5.6.39 Multiple vulnerabilities
HIGH	7.5	142591	PHP < 7.3.24 Multiple Vulnerabilities
HIGH	7.5	42873	SSL Medium Strength Cipher Suites Supported (SWEET32)
HIGH	7.5*	42411	Microsoft Windows SMB Shares Unprivileged Access

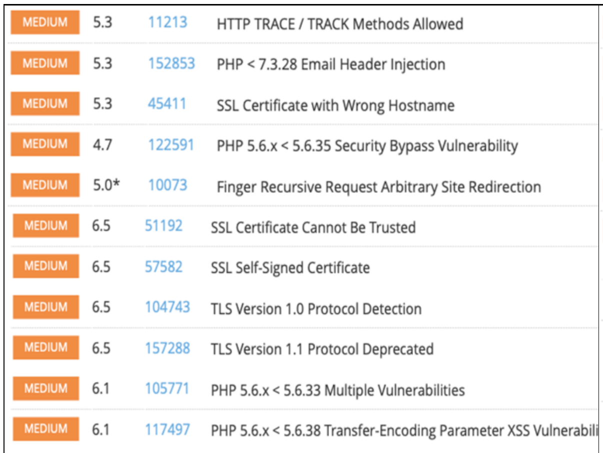
[Figure 20: High vulnerabilities]

High vulnerabilities do not have the same instantaneous widespread impact on essential services of a business when exploited as critical vulnerabilities do. Similarly, neither do high vulnerabilities have the same scope. For example, if a high vulnerability were to be exploited there is less of a chance that this one exploited vulnerability, on its own could be responsible for crashing an entire companies' network or causing other series damage to the company through stopping essential services running. That is not to say that these vulnerabilities could not cause the serious damage mentioned, but often they are less known to malicious actors, or it might require more than one vulnerability to have the same result as one critical vulnerability.

Much like the critical vulnerability results, the high results also highlighted that there were multiple versions of PHP running that are affected by multiple vulnerabilities on '*PHP 5.6.x < 5.6.36 Multiple Vulnerabilities*'. However, redirecting focus is a high rated vulnerability discovered by Nessus, the '*PHP 5.6.x < 5.6.37 exif_thumbnail_extract() DoS*'. This vulnerability is listed on the Nessus database as a denial-of-service vulnerability. This denial-of-service vulnerability is designed to interrupt or disable the web service functioning as normal. Another vulnerability scan with CVE 2018-14851 has this vulnerability as being able to be exploited through a crafted JPEG file. This type of action can show the wide variation in how these vulnerabilities could be exploited. Therefore, this can make it essential to follow the suggestions that are listed on the Nessus database, such as upgrading to a version of PHP that is currently supported. This type of file manipulation to gain access to an exploit will be looked at in further depth in the system hacking section as it will discuss the use of the PHP reverse shell to enable an exploit, this shell was created through bypassing filtering, making one file appear like it is another,

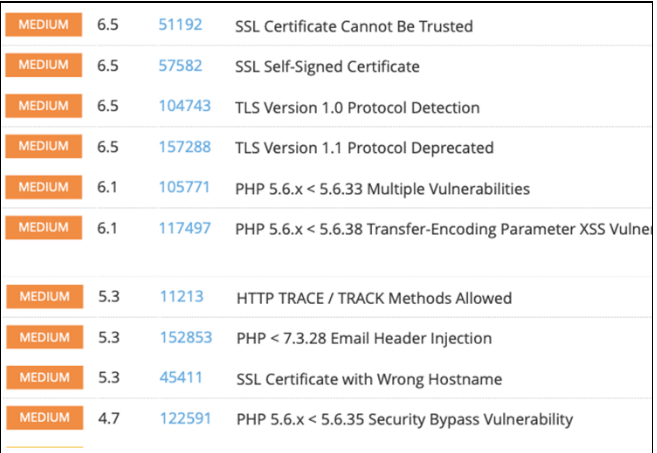
Finally, Microsoft Windows SMB Shares Unprivileged Access can be an important vulnerability for someone to use to gain access to the company network. This is because, when compared to previous vulnerabilities, there is less technological knowledge required to be able to cause damage through this vulnerability. For example, this vulnerability covers the idea that an individual can access multiple areas of the network without having to alter the credentials or permissions of the account, regardless of the level the account is at.

2.4.4 Medium Vulnerabilities



MEDIUM	5.3	11213	HTTP TRACE / TRACK Methods Allowed
MEDIUM	5.3	152853	PHP < 7.3.28 Email Header Injection
MEDIUM	5.3	45411	SSL Certificate with Wrong Hostname
MEDIUM	4.7	122591	PHP 5.6.x < 5.6.35 Security Bypass Vulnerability
MEDIUM	5.0*	10073	Finger Recursive Request Arbitrary Site Redirection
MEDIUM	6.5	51192	SSL Certificate Cannot Be Trusted
MEDIUM	6.5	57582	SSL Self-Signed Certificate
MEDIUM	6.5	104743	TLS Version 1.0 Protocol Detection
MEDIUM	6.5	157288	TLS Version 1.1 Protocol Deprecated
MEDIUM	6.1	105771	PHP 5.6.x < 5.6.33 Multiple Vulnerabilities
MEDIUM	6.1	117497	PHP 5.6.x < 5.6.38 Transfer-Encoding Parameter XSS Vulnerabili

[Figure 21: Medium Vulnerabilities – Sever 192.168.10.1].



MEDIUM	6.5	51192	SSL Certificate Cannot Be Trusted
MEDIUM	6.5	57582	SSL Self-Signed Certificate
MEDIUM	6.5	104743	TLS Version 1.0 Protocol Detection
MEDIUM	6.5	157288	TLS Version 1.1 Protocol Deprecated
MEDIUM	6.1	105771	PHP 5.6.x < 5.6.33 Multiple Vulnerabilities
MEDIUM	6.1	117497	PHP 5.6.x < 5.6.38 Transfer-Encoding Parameter XSS Vulne
MEDIUM	5.3	11213	HTTP TRACE / TRACK Methods Allowed
MEDIUM	5.3	152853	PHP < 7.3.28 Email Header Injection
MEDIUM	5.3	45411	SSL Certificate with Wrong Hostname
MEDIUM	4.7	122591	PHP 5.6.x < 5.6.35 Security Bypass Vulnerability

[Figure 22: Medium Vulnerabilities - Server 192.168.10.2]

The medium results were separated into the two servers; however, they share several **SIMILAR VULNERABILITIES** such as most of the vulnerabilities listed are primarily being centred around web server vulnerabilities. Following on from critical & high, it can be assumed that it is a similar step down regarding the severity of the exploit which the vulnerabilities point too, this is particularly true regarding the scope & extent to the damage can have on Company X. HTTP TRACE/ TRACK methods allowed is an interesting vulnerability to take note of. These TRACE and TRACK are HTTP methods that are used to debug server connections. These can provide information about the webservice potentially leading to accidental disclosure of sensitive information. PHP < 7.3.28 Email Header Injection allows remote attackers to gain control of email header content through an email header injection. This means that attackers can store malicious files or code within this header and send it around the company hoping to infect those who open the email. These vulnerabilities, which not focused on essential network architecture of the business, remain a potential option for an online attacker to gain some level of access to the companies' network.

SYSTEM HACKING

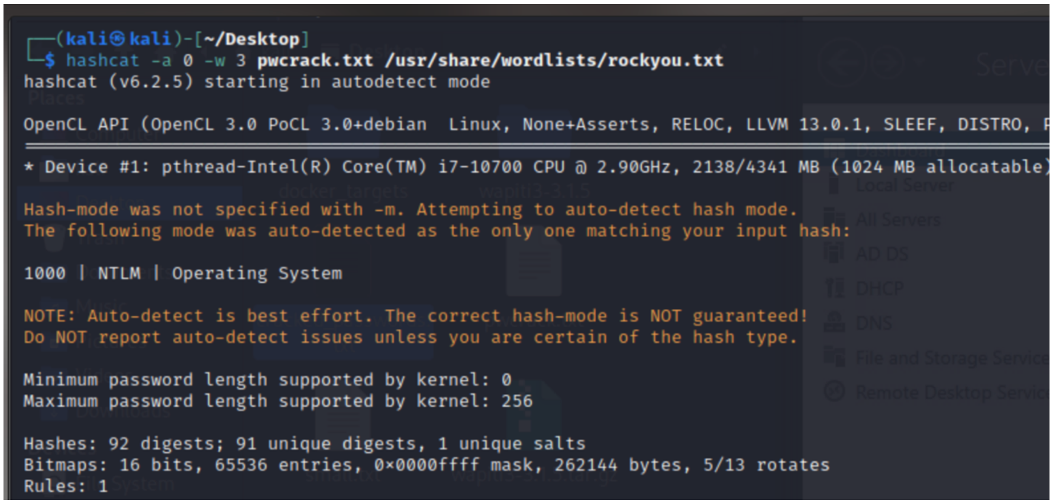
2.4.5 Exploit 1 – 'Rejetto'

The first exploit was discovered during the port scanning phase of the penetration testing which required utilising the scanning capabilities of Nmap. When scanning the TCP ports, port 80 was discovered to be a HTTP File Server running version 2.3. This file server can be visited in the browser and was determined to be advantageous location to start gathering information on the network. Quite quickly into the research of the HTTPFileSever it was discovered that a well-known exploit existed for this type of server, running this version. This is confirmed by cross referencing it with Exploit DB where it was found that this exploit was entitled Rejetto HTTP File Server (HFS) 2.3.x - Remote Command Execution (2) (Thapa, 2016).

After learning of the name of the exploit, Metasploit was used to search for Rejetto. The Metasploit database had this known exploit, therefore meterpreter options were configured to enable this exploit to be usable on the target ip address of 192.168.10.1:80.

2.4.5.1 Password Cracking with Hash Cat

Once conditions were satisfied on Metasploit the exploit was able to be utilised, gaining access to the server through a meterpreter session. From here, the hash dump command was able to be run on the server retuning a list of users & their corresponding hashes. These hashes can then be used to form the basis of a dictionary attack with a required wordlist. This dictionary attack was performed on HashCat. Hash Cat was chosen due to its ability to process the passwords quicker than some other tools such as John the Ripper.

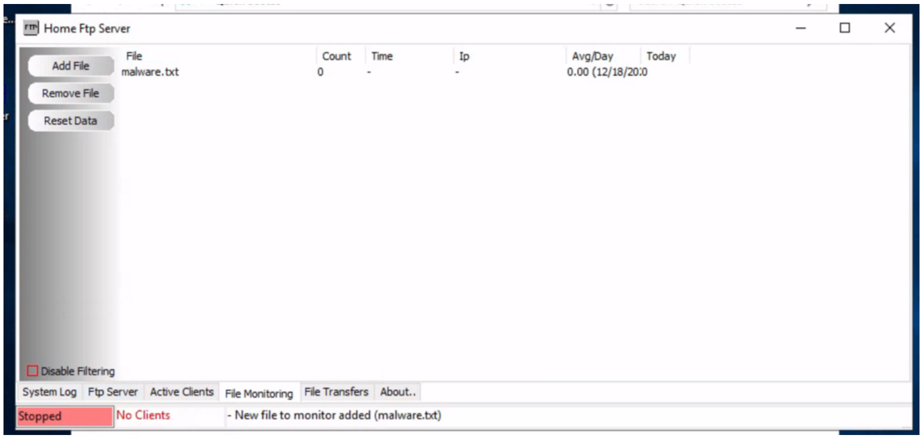


```
┌──(kali㉿kali)-[~/Desktop]
└─$ hashcat -a 0 -w 3 pwcrack.txt /usr/share/wordlists/rockyou.txt
hashcat (v6.2.5) starting in autodetect mode

OpenCL API (OpenCL 3.0 PoCL 3.0+debian  Linux, None+Asserts, RELOC, LLVM 13.0.1, SLEEF, DISTRO, P
==================================================================================================
* Device #1: pthread-Intel(R) Core(TM) i7-10700 CPU @ 2.90GHz, 2138/4341 MB (1024 MB allocatable)

Hash-mode was not specified with -m. Attempting to auto-detect hash mode.
The following mode was auto-detected as the only one matching your input hash:

1000 | NTLM | Operating System

NOTE: Auto-detect is best effort. The correct hash-mode is NOT guaranteed!
Do NOT report auto-detect issues unless you are certain of the hash type.

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 256

Hashes: 92 digests; 91 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0×0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1
```

[Figure 22 : Using Hash Cat to crack passwords]

In total, 15 passwords were cracked. Two of these cracked passwords belong to domain admins. The name of the domain admins are known as a result of enumeration carried out in the previous section. These two admins are *W.Holt & M.Padillia*. The login of W.Holt was used as it was also learnt in enumeration that W.Holt is also the DNSAdmin provide this account with more administrative power.

A program called xfreeRDP was used to gain Remote Access to the server using W.Holts login. From this login, there was a multitude for accessible options to be executed. Take the following example, the FTP file server manager was stored on here, without additional protection. This allowed for files to be uploaded directly to the server. Take this malware.txt as an example of the full admin rights that were accessible. For more information on the services accessible see appendix D for further images.



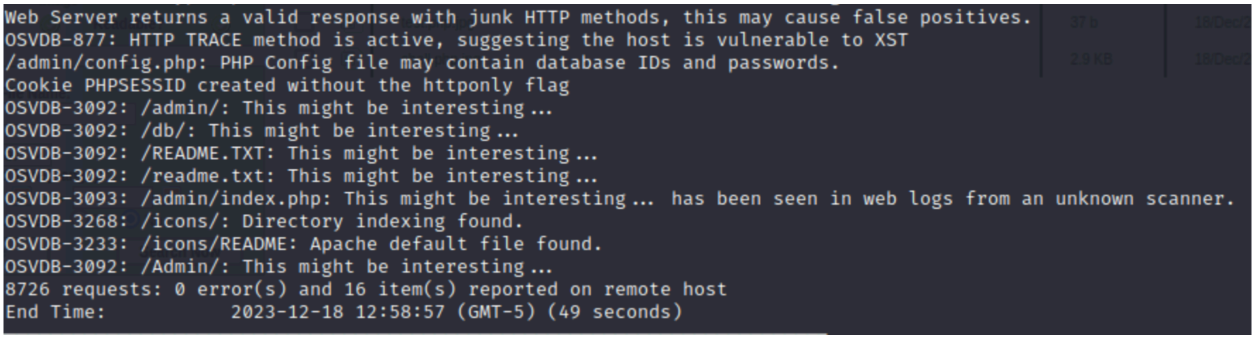
File	Count	Time	Ip	Avg/Day	Today
malware.txt	0	-	-	0.00 (12/18/20:0	

Stopped | No Clients | - New file to monitor added (malware.txt)

[Figure 23: Using FTP file server to move files - example]

2.4.6 Exploit 2 – 'PHP Reverse Shell'

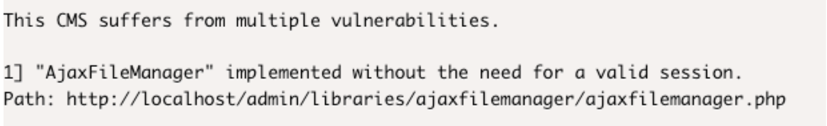
When carrying out scans on interesting ports using Nikto, there was several interesting files found on 192.168.10.1:90. These files are listed in the figure below. When selecting the readme.txt some interesting information was found regarding log1.CMS.



```
Web Server returns a valid response with junk HTTP methods, this may cause false positives.
OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
/admin/config.php: PHP Config file may contain database IDs and passwords.
Cookie PHPSESSID created without the httponly flag
OSVDB-3092: /admin/: This might be interesting...
OSVDB-3092: /db/: This might be interesting...
OSVDB-3092: /README.TXT: This might be interesting...
OSVDB-3092: /readme.txt: This might be interesting...
OSVDB-3093: /admin/index.php: This might be interesting... has been seen in web logs from an unknown scanner.
OSVDB-3268: /icons/: Directory indexing found.
OSVDB-3233: /icons/README: Apache default file found.
OSVDB-3092: /Admin/: This might be interesting...
8726 requests: 0 error(s) and 16 item(s) reported on remote host
End Time:           2023-12-18 12:58:57 (GMT-5) (49 seconds)
```

[Figure 24: Nikto scan]

Upon navigating to log1.cms on the web browser it was clear there was something suspicious that required extra research. Therefore, upon searching for potential exploits related to log1.cms, Exploit.DB had large amounts of information surrounding log1.cms. The following process was trial and error, until falling on the directory listed in the figure below. This directory takes the user to a ajax file manager where the user appeared to be able to upload files to this log1.CMS.



```
This CMS suffers from multiple vulnerabilities.

1] "AjaxFileManager" implemented without the need for a valid session.
Path: http://localhost/admin/libraries/ajaxfilemanager/ajaxfilemanager.php
```

[Figure 25 – Source; https://www.exploit-db.com/exploits/16969]

Trying to upload files proved unsuccessful but there was already a JPG file uploaded to the file manager. Upon further testing. Files with a JPG extension were the acceptable file type for upload. Having the ability to upload files at all to any page on the webserver, which at this point required no cracked password, all it required was the correct scan and some research, makes this an extremally dangerous vulnerability which means anyone with a scanner can access important file managers on Company X's server 1.

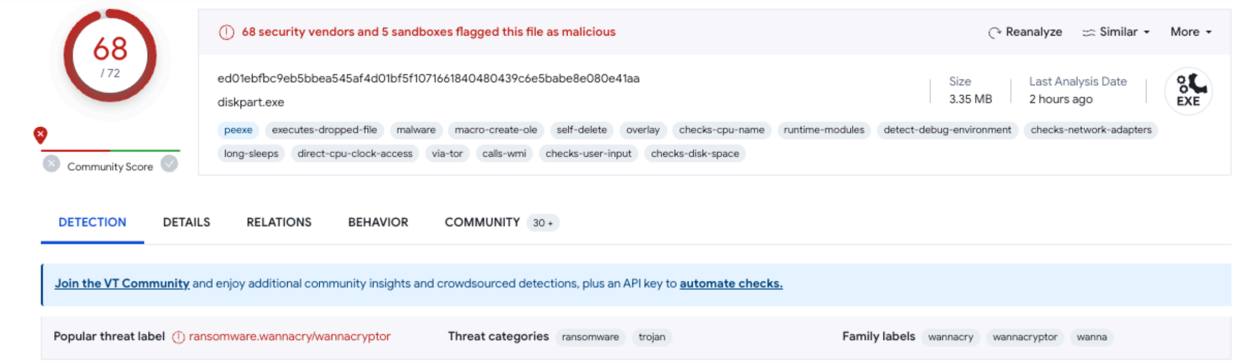
A common exploit often used by remote attackers is this method of file hiding, it can allow upload filtering to be manipulated meaning the restrictions put into place before no longer is relevant. For this example, it was tested with a PHP ‘Hello World’ file first to see if it was possible to hide a PHP file as a .jpg. As it was possible, this meant it would be possible to hide payloads by appending .jpg to their file extension. This camouflage enabled the successful upload of the PHP file. From here exploitation was done with msfvenom on the Metasploit framework. Msfvenom was used to craft a non meterpreter web payload in PHP format. By doing so this allowed the payload to be suitable for this target server. Through running this exploit execution of the PHP shell was granted and so was the unauthorised access to the Apache Server on 192.168.10.1

2.5 Malware Analysis

Ensuring the malware was sandboxed in the virtual machine is crucial to ensure it does not get out to infect the host computer if the malware happens to be executed. When discovering the malware within the sandboxed environment it was important to handle the malware with correct procedure, despite it being sandboxed, this is just to ensure consistency for all future analysis.

2.5.1 Static Analysis

Firstly, an MD5 Hash was taken of the malware. Using this MD5 Hash, it then allowed for a virus total scan to be done to check if there was any matches. When doing so, there was a resounding match for the ‘WannaCry’ ransomware. This ransomware has been the result of many large breaches with the UK (Scaife, Traynor and Butler, 2017) so it is possible to cross check the analysis of WannaCry with the information found from other sources to ensure accuracy. The MD5 has of this file was *84C82835A5D21BBCF75A61706D8AB549*



68 security vendors and 5 sandboxes flagged this file as malicious

ed01ebfbc9eb5bbea545af4d01bf5f1071661840480439c6e5babe8e080e41aa
diskpart.exe

Size
3.35 MB

Last Analysis Date
2 hours ago

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

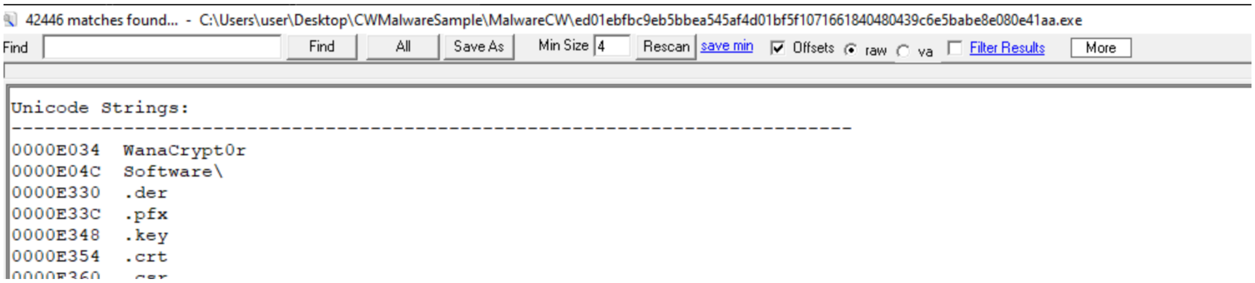
Popular threat label ransomware.wannacry/wannacryptor

Threat categories ransomware trojan

Family labels wannacry wannacryptor wanna

[Figure 26: Source - https://www.virustotal.com/gui/file/ed01ebfbc9eb5bbea545af4d01bf5f1071661840480439c6e5babe8e080e41aa]

It is possible that the remote attacker who managed to get this malware onto the system tried to use some sort of polymorphic malware to obfuscate another piece of malware with elements of the WannaCry ransomware. This can sometimes be attempted by remote attackers as It can send those looking to remove the virus down wrong avenues whilst the real malware has time to develop its malicious activities. However, this was easily debunked by running strings on the file. When running strings on the .exe file it is indeed WannaCry. This can be shown in the multiple references to WannaCry in the strings output as seen in the figure below.



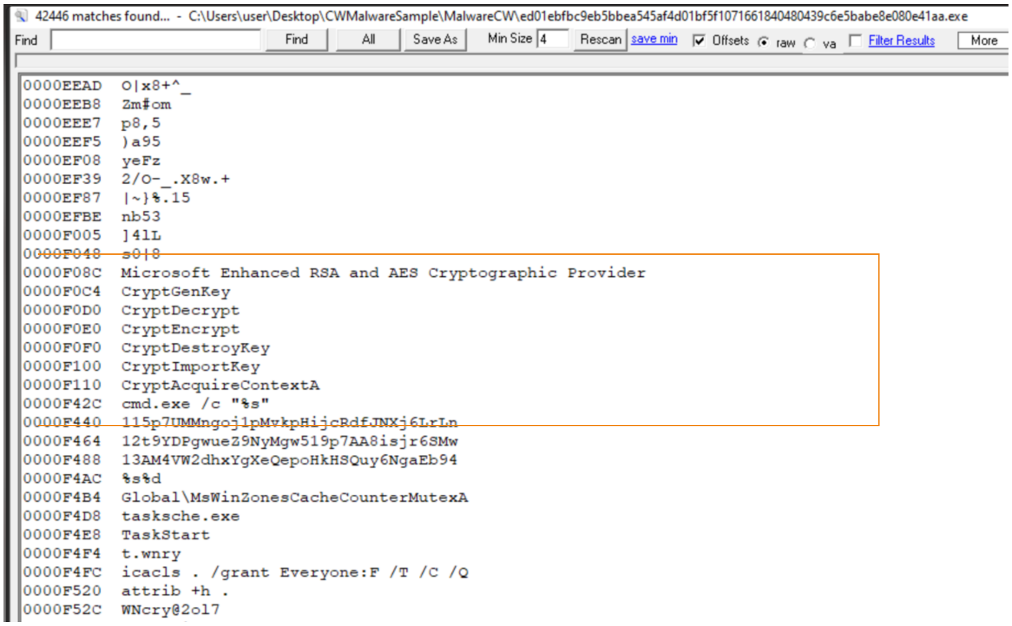
42446 matches found... - C:\Users\user\Desktop\CWMalwareSample\MalwareCW\ed01ebfbc9eb5bbea545af4d01bf5f1071661840480439c6e5babe8e080e41aa.exe

Min Size 4 ☑ Offsets ● raw ○ va □ Filter Results

```
Unicode Strings:
---------------------------------------------------------------------------
0000E034  WanaCrypt0r
0000E04C  Software\
0000E330  .der
0000E33C  .pfx
0000E348  .key
0000E354  .crt
0000E360  .csr
```

[Figure 27: Strings on WannaCry]

Additionally, when looking at the strings output, there are numerous references to the type of activities this file will take when executed. Firstly, there is a list of all known file extensions, it is very rare that executables reference every file extension. Therefore, it can be determined that whatever the executable does, it involves files with every extension. Secondly, looking at the additional strings output there is multiple references to cryptographic providers. This reference to the cryptographic providers shows the encryption method WannaCry uses for locking the user out of all their files upon execution of the ransomware.



42446 matches found... - C:\Users\user\Desktop\CWMalwareSample\MalwareCW\ed01ebfbc9eb5bbea545af4d01bf5f1071661840480439c6e5babe8e080e41aa.exe
Find
Find
All
Save As
Min Size 4
Rescan
save min
☑ Offsets
● raw
○ va
□ Filter Results
More

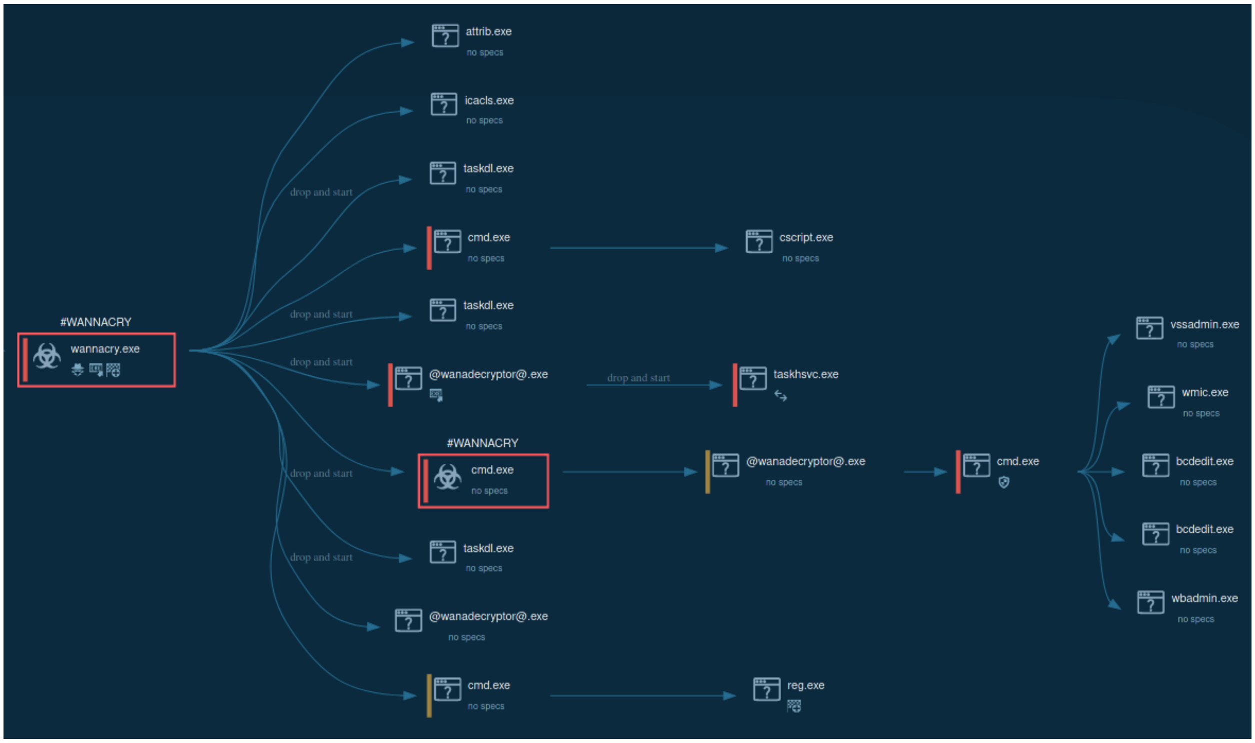
```
0000EEAD  O|x8+^_
0000EEB8  Zm#om
0000EEE7  p8,5
0000EEF5  )a95
0000EF08  yeFz
0000EF39  2/O-_.X8w.+
0000EF87  |~}%.15
0000EFBE  nb53
0000F005  ]41L
0000F048  s0|8
0000F08C  Microsoft Enhanced RSA and AES Cryptographic Provider
0000F0C4  CryptGenKey
0000F0D0  CryptDecrypt
0000F0E0  CryptEncrypt
0000F0F0  CryptDestroyKey
0000F100  CryptImportKey
0000F110  CryptAcquireContextA
0000F42C  cmd.exe /c "%s"
0000F440  115p7UMMngoj1pMvkpHijcRdfJNXj6LrLn
0000F464  12t9YDPgwueZ9NyMgw519p7AA8isjr6SMw
0000F488  13AM4VW2dhxYgXeQepoHkHSQuy6NgaEb94
0000F4AC  %s%d
0000F4B4  Global\MsWinZonesCacheCounterMutexA
0000F4D8  tasksche.exe
0000F4E8  TaskStart
0000F4F4  t.wnry
0000F4FC  icacls . /grant Everyone:F /T /C /Q
0000F520  attrib +h .
0000F52C  WNcry@2ol7
```

[Figure 28: Further Strings on cryptography used]

2.5.2 Dynamic Analysis

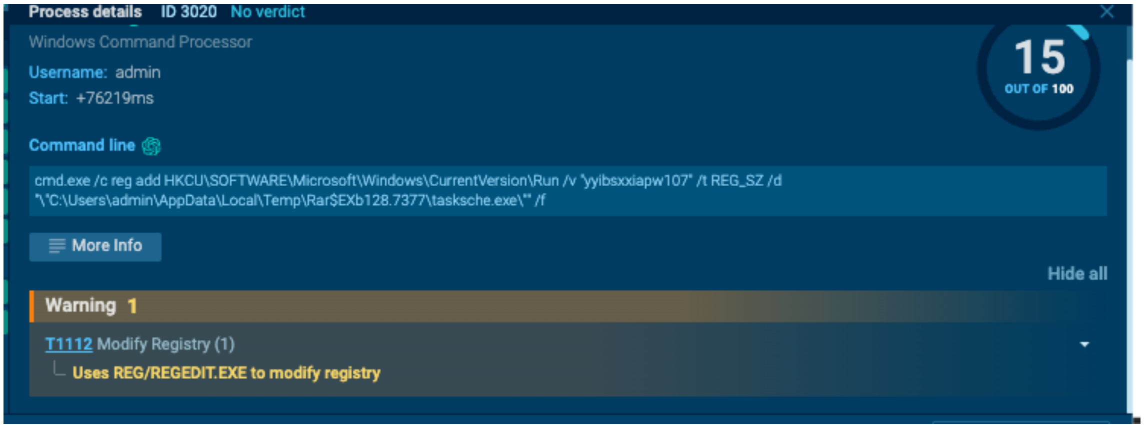
In addition to static analysis of malware it is important to carry out dynamic analysis. Dynamic analysis of malware is typically done by running the malware in a controlled environment, such as the sandbox environment that has already been set up. By running the malware, it is possible to watch its functionality after it has been executed. When putting the malware into the online Application 'App.Any.Run' some interesting information can be extracted from how this malware runs.

When analysing WannaCry in this application, it can be learned that WannaCry tries to exploit the SMB Vulnerability Eternal Blue. Eternal blue acts as the main gateway for WannaCry to infect all nearby systems it can get access to as it utilises the SMB protocol to propagate over the network. This worked for all systems that that did not have up to date patches for SMB. This could be one of the ways in which the malware ended up on Company X's network.



[Figure 29: Execution route of WannaCry - https://any.run/malware-trends/images/wannacry_pg.png]

One of the reasons WannaCry is seen as such a deadly piece of malware is the extensive nature of the malicious activities the malware carries out. For example, WannaCry modifies registry keys In order to hinder any sort of recovery efforts analysts might try to do to counter act the virus. WannaCry looks to make changes in two places in the registry, one of which can be seen in the blow figure. These two locations in registry are **\HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run** as displayed in the figure below, and **\HKCU\Control Panel\ Desktop\.**


Process details ID 3020 No verdict

Windows Command Processor

Username: admin
Start: +76219ms

15 OUT OF 100

Command line

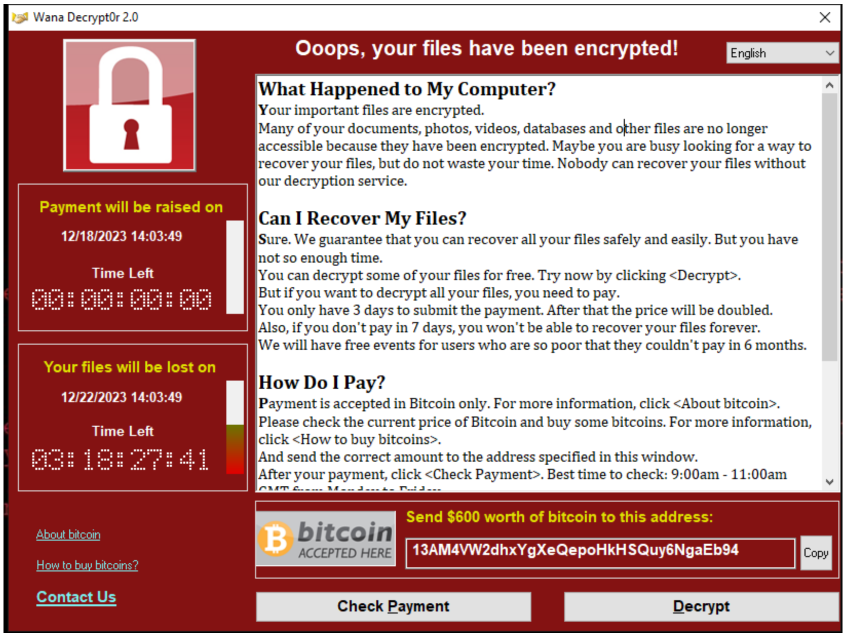
```
cmd.exe /c reg add HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run /v "yyibsxxiapw107" /t REG_SZ /d
"\"C:\Users\admin\AppData\Local\Temp\Rar$EXb128.7377\tasksche.exe\"" /f
```

Warning 1

T1112 Modify Registry (1)
Uses REG/REGEDIT.EXE to modify registry

[Figure 30: Registry key being changed by WannaCry]

When triggered, WannaCry will activate a .exe file called 'Wana Decrypt0r 2.0'. This .exe is the application which presents the unfortunate users with the encryption screen, informing them that they no longer have access to their files. On this screen there is also a bitcoin wallet and instructions on how to get their files back.



Ooops, your files have been encrypted!

Payment will be raised on
12/18/2023 14:03:49
Time Left
00:00:00:00

Your files will be lost on
12/22/2023 14:03:49
Time Left
03:18:27:41

What Happened to My Computer?

Your important files are encrypted.
Many of your documents, photos, videos, databases and other files are no longer accessible because they have been encrypted. Maybe you are busy looking for a way to recover your files, but do not waste your time. Nobody can recover your files without our decryption service.

Can I Recover My Files?

Sure. We guarantee that you can recover all your files safely and easily. But you have not so enough time.
You can decrypt some of your files for free. Try now by clicking <Decrypt>.
But if you want to decrypt all your files, you need to pay.
You only have 3 days to submit the payment. After that the price will be doubled.
Also, if you don't pay in 7 days, you won't be able to recover your files forever.
We will have free events for users who are so poor that they couldn't pay in 6 months.

How Do I Pay?

Payment is accepted in Bitcoin only. For more information, click <About bitcoin>.
Please check the current price of Bitcoin and buy some bitcoins. For more information, click <How to buy bitcoins>.
And send the correct amount to the address specified in this window.
After your payment, click <Check Payment>. Best time to check: 9:00am - 11:00am GMT from Monday to Friday

Send $600 worth of bitcoin to this address:
13AM4VW2dhxYgXeQepoHkHSQuy6NgaEb94

[Figure 31: WannaCry]

3 Discussion

3.1 General Discussion

At the beginning of this report, several aims were outlined to help provide some clarity and guidance to the purpose of the report. These aims will be discussed further and analysed against the findings to determine if these were met.

Aim 1

Develop a comprehensive report on the vulnerabilities that exist on company X's network.

The penetration test of Company X's network encompassed a variety of different techniques, processes & delivered a plethora of security related information for Company X to consider. The methodology behind the penetration test was broken down into understandable, detailed sections that allows the reader to easily identified the process used & the type of information that was obtained. This information was presented through the figures throughout the document and can be used to help replicate the techniques for future testing.

Aim 2

Discover valid exploits and determine the extent to which Company X is exposed to the vulnerabilities discovered during the scanning phases.

Two exploits were discovered and performed throughout this penetration test. The first of the exploits executed was the Rejetto HTTP File Server (HFS) 2.3.x - Remote Command Execution (2). This exploit allowed remote access to the server under domain admin *W.Holt.* This remote access allowed for complete access to the server as an administrator and was able to access web services such as FTP mail server.

Aim 3

Include discussion of the results and develop a set of countermeasures to improve their security going forward.

Each section of the results methodology & discussed the vulnerabilities found how this can be of a potential threat to company X, both in the present and in the feature. Countermeasures are outlined in the next section and should be followed by company X, along with the solutions suggested through the Nessus Scans. This penetration report has allowed for the information to be presented in a understandable way and it will be the responsibility of the company to follow through on these suggestions to help improve its overall security.

Aim 4

Analyse the malware that is sandboxed on their system, identify how this malware operates, what is & how it got onto the network.

Static Malware analysis revealed that the malware sandboxed on company X's machine is WannaCry Ransomware. This was identified through virus total & use of strings to determine key information obtained from the file. Additionally, the malware was opened in a sandboxed environment to observe its functions. The dynamic analysis was carried out in 'App Any Run' and delivered an interesting look into how the malware functions.

Overall, the original aims of the report were delivered through the report, giving an insight into scanning & enumerating to find out information about the network used by company X. The report also fully discusses identifying vulnerabilities, their potential exploits and exploiting some found vulnerabilities which has led to password cracking.

3.2 COUNTERMEASURES

3.2.1 Training & Education

Firstly, before any countermeasures can be firmly put into place correct training and education must be given to the employees of Company X. Without a good foundation of security knowledge through the company, it can be highly difficult for individuals to develop.

3.2.2 Upgrading Legacy Software

During the Nessus scans in the procedure sections, the most common vulnerabilities were coming from the user of old PHP versions. Some of these PHP versions were no longer supported and therefore they are some of the best targets for remote attackers.

3.2.3 Removing Redundancy on servers

When scanning the ports on both servers, it began to become prevalent that there were several shared services across both servers. This is a vulnerability in network security. Having the same type of services on both servers will allow online attackers who gain access to one server using a certain exploit, for example the Rejetto exploit to then use this again on the other server, granting access to the full network.

3.2.4 Password Policy

As discussed during the enumeration phase, there is several password policies missing that would help deter remote attackers from launching brute force attacks like those seen In the Rejetto exploit in the system Hacking section.

3.2.5 Privilege Management

Implement a principle of least privilege (PoLP) throughout the company. This principle will allow for the minimum number of privileges be given to each user for their role. This can allow for users accidentally receiving administrative privileges, or individuals receiving higher privileges than normal due to incorrect configuration. This principle will ensure employees only have access to specific information, tools & data.

3.2.6 Firewalls

When identifying the services on the ports, there were multiple file transfer, data transfer & email services detected. These services involve the exchange require data to be sent from one client to another, for this reason they necessitate a well configured firewall on these services. A firewall can act as a barrier between incoming and outgoing traffic to ensure no malicious activity tries to interfere., implementing a correctly configured firewall on these services would be beneficial. Additionally, this will help provide a counter measure for future attempts at malicious software being implemented on the machine.

3.3 FUTURE WORK

Future work can be carried out on Company X in the form of further exploit analysis. Nessus & port scanning developed a large bank of information on the vulnerabilities that exist on company X's network, therefore it would be extremely beneficial to determine where these vulnerabilities could turn into exploits and help prevent this from happening. Also, to examine the ports in further detail would be very beneficial to gain more of an understanding of how the network architecture is working.

Finally, developing a deeper understanding of the WanaCry malware and if there is a way to determine how this malware ended up on the company's system, if it wasn't through the previous suggested way of the FTP fileserver.

REFERENCES

Alhayani, B.S.A., Abbas, S., Khutar, D.Z. and Mohammed, H.J. (2021) 'Best ways computation intelligent of face cyber attacks', *Materials Today: Proceedings,* . doi: 10.1016/J.MATPR.2021.02.557 https://consensus.app/papers/best-ways-computation-face-cyber-attacks-alhayani/6e084c933ff05be3a4713574b10fd46f/

Jiang, G. (2002) 'Multiple vulnerabilities in SNMP', *Computer (Long Beach, Calif.),* 35(4), pp. supl2-supl4. doi: 10.1109/MC.2002.1012421 https://ieeexplore.ieee.org/document/1012421

Motero, C.D., Higuera, J.R.B., Higuera, J.B., Montalvo, J.A.S. and Gomez, N.G. (2021) 'On Attacking Kerberos Authentication Protocol in Windows Active Directory Services: A Practical Survey', *IEEE access,* 9, pp. 109289-109319. doi: 10.1109/ACCESS.2021.3101446 https://ieeexplore.ieee.org/document/9501961

Naik, N., Kurundkar, G.D., Khamitkar, S. and Kalyankar, N. (2009) 'Penetration Testing: A Roadmap to Network Security', *ArXiv,* abs/0912.3970https://consensus.app/papers/penetration-testing-roadmap-network-security-naik/89a555b575d25733836e5bae019461f1/

Scaife, N., Traynor, P. and Butler, K. (2017) 'Making Sense of the Ransomware Mess (and Planning a Sensible Path Forward)', *IEEE potentials,* 36(6), pp. 28-31. doi: 10.1109/MPOT.2017.2737201 https://ieeexplore.ieee.org/document/8103104

Sebastian, G. (2022) 'Cyber Kill Chain Analysis of Five Major US Data Breaches: Lessons Learnt and Prevention Plan', *International journal of cyber warfare and terrorism,* 12(1), pp. 1-15. doi: 10.4018/IJCWT.315651 http://services.igi-global.com/resolvedoi/resolve.aspx?doi=10.4018/IJCWT.315651

Tervoort, T., De Oliveira, M.T., Pieters, W., Van Gelder, P., Olabarriaga, S.D. and Marquering, H. (2020) 'Solutions for Mitigating Cybersecurity Risks Caused by Legacy Software in Medical Devices: A Scoping Review', *IEEE access,* 8, pp. 84352-84361. doi: 10.1109/ACCESS.2020.2984376 https://ieeexplore.ieee.org/document/9050776

Thapa, A. (2016) *Rejetto HTTP File Server (HFS) 2.3.x - Remote Command Execution (2).* Available at: https://www.exploit-db.com/exploits/39161 (Accessed: Dec 20, 2023)

Weir, M., Aggarwal, S., Collins, M. and Stern, H. (2010) 'Testing metrics for password creation policies by attacking large sets of revealed passwords', *Proceedings of the 17th ACM conference on Computer and communications security,* , pp. 162-175. doi: 10.1145/1866307.1866327 http://dl.acm.org/citation.cfm?id=1866327

Windows (2022) *Maximum password age - Windows Security.* Available at: https://learn.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/maximum-password-age (Accessed: Dec 20, 2023)