

|  |
| --- |
| **Penetration test of a virtual windows network.**  **Isaac Potts**  CMP210: Penetration Testing  2023 |

Abstract (not more than around 400 words)

This paper contains the methodology, tools, and exploits that were used to hack into a virtual network of 2 Windows servers and a client computer. The aim of this paper is to show the risk to this company from a malicious insider. This has been done by gaining full access to the network. It will also provide potential fixes to the discovered vulnerabilities. The penetration tester had been given a standard account with user privileges to conduct their tests.

The penetration tester followed a methodology involving 4 primary steps, footprinting, scanning, enumeration and system hacking. The initial scan and enumeration yielded some promising results, including a vulnerable FTP server and multiple http web pages hosted on nonstandard ports. Initial enumeration gave the penetration tester a viable username using enum4linux and rpcclient. This was followed by directory enumeration, and default credentials discovered for an admin login for log1 CMS were discovered, hosting vulnerable applications.

On server 1, the penetration discovered a log1 cms site hosted on port 90, and an Argosoft mail server running on port 80 that used Finger, pop3 and SMTP ports for communication. Exploits for this were discovered, however, the tester was unable to perform successful exploitation due to the vagueness with which the exploit was written. Server 1 on port 2025 ran a HttpFileServer, which allowed exploitation via a Metasploit module.

On server 2, web pages on port 90 and 2033 were discovered. On port 90, a BoZon file Server was running, with which the penetration tester had some success in exploiting, including adding a user account which was automatically given superuser permissions. On port 2033 HttpFileServer was running which again resulted in exploitation via a Metasploit module.

From the results gathered it was concluded that the virtual network had many harmful vulnerabilities, and that a malicious insider could easily gather a large amount of data and credentials from this network.

Contents

[1 Introduction 1](#_Toc153806934)

[1.1 Background 1](#_Toc153806935)

[1.2 Aim 1](#_Toc153806936)

[2 Procedure 2](#_Toc153806937)

[2.1 Overview of procedure 2](#_Toc153806938)

[2.2 Scanning 2](#_Toc153806939)

[2.3 Enumeration 5](#_Toc153806940)

[2.4 Exploitation 9](#_Toc153806941)

[3 Discussion 17](#_Toc153806942)

[3.1 General Discussion 17](#_Toc153806943)

[3.2 Countermeasures 18](#_Toc153806944)

[3.3 Future Work 19](#_Toc153806945)

[References 20](#_Toc153806946)

[Appendices 22](#_Toc153806947)

[Appendix A 22](#_Toc153806948)

[Appendix B 29](#_Toc153806949)

# Introduction

## Background

Penetration testing is an authorized simulated attack performed on a system to assess its security, by using the same tools that a malicious actor would use and then advise the company how to fix the security issues. (synopsys.com, 2021). Companies in certain sectors are also required to have penetration tests conducted on their network due to the sensitive client data that they are handling.

More than 80% of UK organizations were affected by cyber-attacks in the year 2021/2022, and 73% of businesses in the UK suffered from successful or unsuccessful ransomware attacks. (comparitech.com, 2023) It is also estimated that the cost of cybercrime in the UK is around £27 billion a year. (ramsac.com, 2022). It is not just the business first hacked that is affected, but many more organizations relying on that business, be it for energy, finances, or cloud storage, will feel the effect of a cyber-attack. This is called supply chain disruption. As shown from this, penetration tests are hugely important for businesses to keep them operational, and to prevent successful attacks.

The use of technology has become a part of everybody’s life, with the average person globally spending 6 hours and 58 minutes on an internet connected device a day (explodingtopics.com, 2023) most, if not all will also store personal identifiable information on the device. This mass usage of technology makes cybersecurity, and by association penetration tests, more important on a global scale, to prevent unauthorized access to an individual, a company, or a governments information.

## Aim

The aim of this penetration test was to discover how fully the penetration tester, acting as a malicious insider, could compromise the 2 servers on this virtual network, and feedback countermeasures to the company. In this project the penetration tester expected to follow a methodology which consisted of footprinting, scanning, enumeration and system hacking. The penetration tester also expected to find multiple vulnerable applications and hoped to successfully exploit the discovered vulnerabilities.

# Procedure

## Overview of procedure

The procedure used by the penetration tester, as previously mentioned, is made up of 4 steps: footprinting, scanning, enumeration, and system hacking. The penetration tester would usually begin with footprinting, which is using google, Shodan, and other search engines to discover information such as IP addresses, names of employees, and email addresses. But as this was a penetration test running on designated virtual machines, this step was not possible.

Server 1 had IP address 192.168.10.1, and server 2 had IP address 192.168.10.2.

The next step addressed was scanning, which involved using tools to determine open ports on the targets, what services were running on these ports and detect the presence of firewalls. This was achieved using a mix of Nmap scans and pings.

In this methodology, the next step was enumeration which builds upon the information discovered from scanning. Enumeration involves using developer tools such as rpcclient, and tools designed for penetration testing like enum4linux to learn more about a target, and attempt to discover passwords, password policies, and usernames on the targets.

The next step is System hacking. This step involves using information discovered in the previous 3 phases, to attempt to gain access to the targets by exploitation of vulnerable applications.

## Scanning

**Ping**

Ping is a command line tool in both windows and Unix operating systems, that sends an ICMP echo request to a specified IP address and waits for a reply. Ping was used on both servers to ensure that the Penetration tester had configured the virtual network correctly, and that both machines were online. See figure 1 for more information.

A screen shot of a computer

Description automatically generated

*Figure 1: ping from server 1.*

**Nmap**

Nmap is a command line port scanner, used for network discovery and port scanning. (nmap.org, 2021) It has many different scanning and configuration options available, allowing for many varying scans to be used to determine open ports through firewalls. Nmap also has built in scripting options, as well as the option to build custom scripts, used for brute forcing, banner grabbing, and service enumeration. This was used with banner grabbing scripts on both servers, using a basic TCP scan. Initially, nmap was ran with multiple different flags as shown in Figure 2.



*Figure 2: nmap syntax.*

The command sudo nmap, means run nmap with elevated privileges, which is needed for TCP scans. The -p- flag means to scan all 65536 ports, to look at services running. The -vv flag, means very verbose, meaning the scan will output more information, including errors which is useful for determining additional information about the targets. The -T5 flag is to set a timing template, numbered from T0 – T5, meaning T5 is the fastest possible. The -O flag enables OS detection, and –osscan-guess guesses the operating system more aggressively. The –script=banner tells nmap to run banner grabbing scripts.

Another scan was also ran with the -A flag which ran OS detection, version detection, script scanning and traceroute.

**Server 1 Nmap**

As shown in Appendix A many interesting results were discovered in the server 1 nmap scans. FTP on port 21 hosts a FTP server called Home FTP server. The nmap scan with -A flag it was discovered this server allowed anonymous login. The hostkey for SSH on port 22 was also discovered.

Port 80 hosts a http web mail server called Argosoft mail server version 1.8 (1.8.2.9). This also used pop3 on port 110 and port 25 for smtp. On port 90, a log1 cms web page was discovered. On the initial nmap scan this showed up as dnsix, not http. On port 2025 a HttpFileServer version 2.3 was running. This was incorrectly identified by nmap as ellpack. Multiple windows Active Directory ports were also discovered, giving the penetration tester another attack vector.

**Server 2 Nmap**

As shown in appendix A server 2 (192.168.10.2) had no ftp port open. Server 2 had a BoZon web page running on port 90, and Similar to Server 1, a HttpFileServer version 2.3 was running on port 2025, which was incorrectly identified as glogger on the initial nmap scan. Active directory ports were also open on the target.

**Nessus**

Nessus is a platform developed by tenable that scans for security vulnerabilities in operating systems, applications, devices, and cloud services (techtarget.com, 2023). Nessus was selected to scan this network to automate vulnerability discovery. Nessus also provides an easy-to-read report of the targets, ranking vulnerabilities from critical to low severity. It also provides fixes for the discovered vulnerabilities.

The penetration tester configured Nessus to run using a basic network scan on both servers. Nessus was also given the penetration testers user account credentials for access with SSH. The Nessus scan discovered many issues with the version of PHP on Server 1 which were rated critical. Also, it identified an unauthenticated account creation vulnerability, and cross site scripting vulnerabilities in Argosoft mail server. On Server 2, it was discovered that the NETLOGON SMB share was able to be read, however other vulnerabilities in the web pages were not discovered.

**PingCastle**

PingCastle is another vulnerability scanner, specifically designed to audit Active Directory systems and ranks them on a scale of 0-100. The Penetration tester used PingCastle on the Client1 machine to generate a report on this network’s Active directory. This scan successfully discovered Administrator usernames, as shown in figure 3. Associated emails were also discovered. It also successfully identified other problems with the Active Directory.

Administrator usernames enumerated that were not in special group Protected Users.

A screenshot of a computer

Description automatically generated

*Figure 3: Administrator usernames from PingCastle*

The Anomalies analysis scored 100/100, due to multiple issues such as the Kerberos password not being changed, and LAPS, Local Administrator Password Solution, not being installed. It also revealed the Domain name to be uadcwnet.com, along with a user L.Gill who did not require Kerberos pre-authentication.

## Enumeration

**Enum4linux**

Enum4linux is a command line tool for Linux to automate windows enumeration, and it achieves this through the usage of legitimate developer tools such as smbclient, rpcclient, net, and nmblookup. (github.com, 2021). This was run on both servers with the same flags and got the same results.

This syntax was used on both servers as shown in figure 4.



*Figure 4: enum4linux syntax*

The -a flag means all, as in enumerate everything. The -u flag denotes the username, and the -p flag denotes the password to be used.

Information from the SMB shares were discovered, along with the RID, username and description of all user and administrator accounts. The password policy was discovered, of which the minimum password length was 7 and there was no lockout policy, meaning brute forcing was a viable option. The SIDs of numerous accounts were also discovered, and used to discover if a user was a local user, or part of a Domain Group.

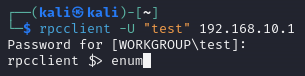
Of particular interest is the description of the user Holly Mclaughlin, containing their password trainmen63 as shown in figure 5.



*Figure 5: Password of Holly McLaughlin*

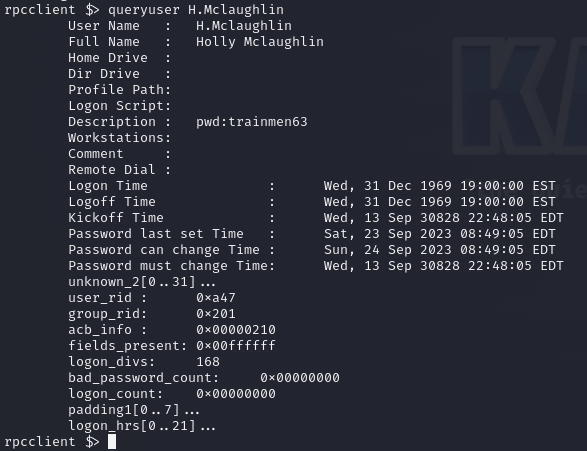
**Rpcclient**

RPC stands for Remote Procedure Call which helps establish and maintain communication between multiple windows application. Rpcclient is used to communicate with this service. (hackingarticles.in, 2021) Enum4linux already uses rpcclient for automatic enumeration, but using it for manual enumeration allows one to gather more information. It was used with the syntax shown in figure 6.



*Figure 6: Rpcclient syntax*

In rpcclient privileged tokens were discovered using enumprivs and found 35 privileges. The user H.Mclaughlin was queried, and the password in their description confirmed, as seen in figure 7.



*Figure 7: Holly Mclaughlin rpcclient description*

Using command querydominfo it was again confirmed that the domain used was UADCWNET, and the createdomuser was used, but the user did not have permissions to create a user or give them a password. See figure 7.



*Figure 8: createdomuser access denied.*

**Winpeas**

Usually, Winpeas would be a post exploitation or exploitation tool as it detects privilege escalation, but it is being included in enumeration as the penetration tester already has user access.

Winpeas is a script that searches for possible paths to escalate privileges on devices with Windows operating systems. It checks for registry permissions, user accounts, access controls, and patch levels (vk9-sec.com, 2022).

It was discovered by the penetration tester that files could be put on the FTP home server on server 1 either by using an anonymous FTP account (username and password both the word anonymous) or the account the penetration tester was given. This also worked with the H.Mclaughlin user account. The Penetration Tester then used ssh to access the machine and execute Winpeas.exe.

WinPeas discovered that the servers may be vulnerable to a privilege escalation exploit, called KrbRelayUp. Current token privileges were also acquired, that could be used to escalate privileges. Services vulnerable to DLL hijacking were then discovered, such as nssm.exe, ssh-agent.exe, and sshd.exe. It also confirmed no domain, private or public firewall was enabled. WinPeas also retrieved the hash for the penetration testers account. However, much of WinPeas abilities were not allowed, due to the Penetration tester not having a privileged account, services not being found and certain PowerShell commands that were disabled.

**PowerShell**

PowerShell is a scripting language used for windows to automate tasks on Windows computers but can also be used maliciously to enumerate and exploit a target. The penetration tester also used ssh to explore the files stored on the target. Both H.Mclaughlins and the test account were used. The Penetration tester discovered it was possible to use PowerShell through ssh on both accounts. Basic PowerShell commands were run, “Get-LocalGroup” to look at local groups, which was successful, and “systeminfo” was attempted to find information on what system was running, but the test user did not have permission to execute this command as shown in figure 9.

A screen shot of a computer

Description automatically generated

*Figure 9: Failed command*

Applications on the server were discovered, using the syntax shown in figure 10.

A screenshot of a computer

Description automatically generated

*Figure 10: Services running on the server.*

These services may be vulnerable to DLL hijacking, as seen by WinPeas.

More commands were run on the server to attempt further enumeration, but no more information than was already discovered was found.

**DNS zone transfer**

A DNS zone transfer is a normal operation between primary and secondary DNS servers to synchronize records for a domain, however, this can sometimes be accessed by an attacker (hackertarget.com, 2023). This was attempted by the penetration tester using the nslookup tool. However, this was unsuccessful as shown in Figure 11. The same result was received using server 2.

A screen shot of a computer

Description automatically generated

*Figure 11: Failed zone transfer*

**Gobuster**

Gobuster is a URL directory brute forcing tool, used for discovering hidden directories and files on an internet application, and is often used to discover login pages, or configuration files. (geeksforgeeks.org, 2023) Gobuster has multiple modes, dir for directory brute forcing, DNS for enumerating subdomains, and vhost for enumerating virtual hosts. The Penetration tester used the dir option, using the syntax shown in figure 12, with -w denoting the path to the wordlist, and -u flag specifying the URL.



*Figure 12: gobuster directory brute forcing syntax.*

**Gobuster server 1**

On server 1, an Argosoft mail server was running. When gobuster was used on this web server, a multitude of errors arose, and this was ultimately unsuccessful.

On port 90 a log1 cms site was running. Gobuster was used on this, discovering a readme.txt and an admin panel, along with a templates, and a lightbox page. See figure 13 for output.

A screen shot of a computer

Description automatically generated

*Figure 13: Gobuster output*

Gobuster was then used on HFS, which was unsuccessful.

**Gobuster server 2**

Gobuster was used on the BoZon web page running on port 90, and a private page was revealed, which contained useful files.

It was again used on HFS to no avail.

## Exploitation

**Exploit identification Server 1**

Server 1 contains 2 webpages, the first of which is an Argosoft Mail Server, which allows a user to create and use an account to send mail to other users. Using searchsploit, 2 exploits were identified. One was a directory traversal, and the other was an authentication bypass. See Figure 14 for more information.

A screen shot of a computer

Description automatically generated

*Figure 14: Argosoft exploits*

However, Directory Traversal only affects versions 1.8.1.5, and the version running is 1.8.2.9, therefore it is not vulnerable.

Multiple vulnerabilities were also discovered on Home FTP Server, see figure 15 for more information.

A computer screen shot of a program

Description automatically generated

*Figure 15: Home FTP vulnerabilities.*

On port 90 a log1 cms webpage was running which allows users to view a basic web page. Upon reading the readme.txt the penetration tester discovered credentials for the admin login, which was user log1 and password log1. See Appendix B for more information. Upon login with the discovered credentials, it was discovered that ajax file manager was running. Multiple exploits were identified for this web page, including a PHP code injection, and a create folder vulnerability. See figure 16 for all vulnerabilities:

A screen shot of a computer

Description automatically generated

*Figure 16: log1 cms exploits*

On port 2033, a HttpFileServer was running, created by Rejetto. It allows a user to login, and upload files. By searching for Rejetto on searchsploit many vulnerabilities were discovered. See figure 14 for a list.

A screenshot of a computer program

Description automatically generated

*Figure 17: HFS exploits*

**Exploit identification Server 2**

Port 90 had a BoZon file upload webpage running, which allowed the penetration tester to create an account, and upload files. This had one Remote code execution vulnerability, which allows users to execute commands or create a new account.

Server 2 has HFS running on port 2025. See previous figure 14 for vulnerabilities.

**Home FTP Server**

Multiple vulnerabilities were also discovered in the Home FTP Server, and a directory traversal vulnerability was successfully utilized, by appending a ../ to the ls command, allowed directory traversal by the penetration tester. This FTP server also allowed anonymous login, meaning anyone could access anything on the filesystem. This vulnerability was used to successfully get a file titled Halloween-History.pdf from HR shares in directory titled Derek. See Figure 18 for syntax.

A black screen with white text

Description automatically generated

*Figure 18: FTP gets Halloween-History.pdf.*

This technique was also used to access the Administrators directory as shown in figure 19.

A screenshot of a computer

Description automatically generated

*Figure 19: lists Administrator user.*

**KrbRelayUp**

KrbRelayUp is a universal no fix privilege escalation vulnerability that requires LDAP signing to be off, and optionally the permissions to add new computers to an Active directory (github.com/Dec0ne, 2022). This was downloaded and compiled using Visual Studio 2022. The build command was then used to turn it into an executable, and ftp with the test account was used to upload the file to Server 1. It was also found that evil-Winrm could be used to put files into any directory that the penetration tester had permission to access. See figure 20 for more information on ftp upload.

A computer screen shot of a black screen

Description automatically generated

*Figure 20: uploading KrbRelayUp via ftp.*

The penetration tester then logged into Server1 using the test account on SSH, and then attempted to execute KrbRelayUp to add a new computer with elevated privileges onto the network. However, every attempt was met with errors, as the Penetration tester did not have required permissions to run specific processes. See figure 21 for exact syntax and error messages.

A black screen with white text

Description automatically generated

*Figure 21: error with KrbRelayUp*

However, the Penetration tester could successfully add a computer, and use the account, just with user privileges, not system privileges as the exploit was meant to give the penetration tester. This was also tried with different methods, but with the same result.

**Server 1 exploitation**

The Argosoft authentication bypass was explored, and the penetration tester experimented with the website, trying different usernames, and changing the settings they were given access to, but they could not discover any other user data.

On port 90, the penetration tester logged in with the discovered username and password from the readme.txt found during enumeration and discovered an Ajax File manager application. This exploit works because the data.php where data is processed performs no filtering on given input. However, despite the Penetration tester changing the port number, and specifying the path which was simply a / as the code already contained the correct path, the exploit was unsuccessful, as seen in figure 22.

A screenshot of a computer

Description automatically generated

*Figure 22: unsuccessful exploit on port 90*

The Metasploit version of this vulnerability was also attempted, using the same syntax but the penetration tester had no success with this exploit. The penetration tester then attempted to upload multiple php files containing reverse shells, and set up a netcat listener on their device, but uploading php files was not allowed by the server. Other extensions such as .phtml, .htaccess, .phps and php2 were used, along with other extensions but none were successful (book.hacktricks.xyz, 2023). The penetration tester then successfully uploaded a .exe reverse shell, but was unable to get the executable to run.

Next the penetration tester attempted to exploit the HttpFileServer, using the Metasploit module rejetto\_hfs\_exec. This Metasploit module works by exploiting a bad regex in the ParserLib.pas, bypassing the filtering by using %00 (exploit-db.com, 2014). It was configured using the syntax shown in figure 23.

A screenshot of a computer code

Description automatically generated

*Figure 23: Http rejetto exploit.*

It was then run with the run command, and exploitation was successful, giving the penetration tester a meterpreter session. The penetration tester then ran the command getuid to discover what permission they were running as, and they were running as NT AUTHORITY\SYSTEM, as shown in figure 24.



*Figure 24: running as system.*

This output means the penetration tester has the highest possible privileges on this network. The hashdump command was then ran, and hashes of all users were collected and saved to a text file for later use.

The penetration tester then attempted to add persistence to the server, using a netcat executable. First the penetration tester checked what runs at boot, as seen in Figure 25.

A computer screen with white text

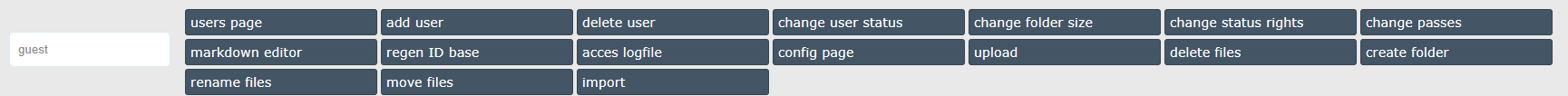
Description automatically generated

*Figure 25: check services at boot*

Following on from this, the nc.exe was then uploaded, and set to start at boot, and a firewall exception was created, however when the server was rebooted the server crashed, resetting it to default settings preventing the penetration tester from creating persistence.

**Server 2 exploitation**

Running on port 90 was a BoZon filesharing webpage. This had a register and login functionality, allowing the penetration tester to create an account, and an account was made with username test and password test123. Once logged in, the penetration tester discovered they had access to the configure profile rights page, allowing the penetration tester to change new profile, guest, user and admin privileges simply by clicking on option, see figure 26 for example of the penetration tester changing guest level permissions.



*Figure 26: changing permissions*

It was also possible to configure BoZon, letting the penetration tester change what files were allowed, change use of lightbox, and change size of folder to upload. The Penetration tester was also able to change other users’ password and delete other accounts and files.

The penetration tester attempted to upload PHP reverse shells but was unable to get the shells to execute on the target. However, file upload of any type is permitted.

The arbitrary code execution vulnerability allowed users to write and store system commands in the file auto\_restrict\_users.php in the private directory of BoZon, discovered by Gobuster in the enumeration section. These commands are then executed when the Penetration tester logs into BoZon or refreshes the page. Alternatively, this also allows for new user creation.

To configure the exploit, the URL variable was changed to the location of index.php, see figure 27 for new URL variable. It was then saved and given the name exploit.py.



*Figure 27: new URL variable*

The exploit was written in python and was ran using python2.

See figure 28 for syntax to run this exploit.

A screen shot of a computer screen

Description automatically generated

*Figure 28: Running the exploit.*

User account with username Apparition and password abc123 was created as shown in figure 29.



*Figure 29: New user created.*

Following on from this, the penetration tester attempted running the phpinfo command on the server, which was executed successfully. This gave the penetration tester much information about the machine, file system and environment on which the web page ran. The penetration tester attempted to get the web page to echo Hello World by replacing the phpinfo command in exploit.py but this crashed the server. A PHP reverse shell was also uploaded, and other commands were attempted but all crashed the server.

On port 2033 HttpFileServer was running. The same process used in compromising the same web page on server 1 was used, except the penetration tester changed the Metasploit variables RHOST and RPORT to match the port and server 2 IP address. This exploit was successful.

The Penetration tester then checked what permissions this was running as, and the meterpreter shell was running with system authority. NTLM hashes of all users were then taken using the Metasploit hashdump command which appeared to be the same as the hashes from server 1. The attacker then explored the filesystem, and attempted to add netcat to the list of services that would run on startup, however this crashed the server, the same as server 1.

**John**

John the ripper is a brute forcing tool designed for cracking hashes and was used by the penetration tester to crack the NTLM hashes. Out of all the users and the 7 administrators, 18 hashes were cracked and 4 administrator accounts passwords were discovered, as indicated by figure 30.

A screenshot of a computer

Description automatically generated

*Figure 30: Administrator passwords*

**RDP**

RDP is shorthand for remote desktop protocol and allows a user remote access to the GUI of a computer. The penetration tester logged onto both Server1 and Server 2 with the W.holt administrator account, but any of the administrator accounts could be used.

On load with RDP both servers loaded with Server Manager which allowed the penetration tester to view, start and stop services running on the server. The Penetration Tester was also able to access multiple files on Server 1 and Server 2 Network shares. These files contained images of the Manchester United logo, images of many different rugby teams and multiple files about Online Video and Word in different formats.

# Discussion

## General Discussion

Through using the methodology mentioned in this report, the Penetration tester was able to compromise the entire network. However, the Penetration Tester did begin with a test account, removing initial access, pivoting and footprinting which significantly shortened the length of the penetration test. By following this methodology, it was quite simple to identify vulnerabilities running on this companies network.

On first exploitation on the FTP Home server, all shares and the Administrator user was able to be accessed using directory traversal, and on first compromise on the HttpFIleServer, it was running as system, meaning that all services were running with full privileges, making exploitation significantly easier for any attacker. This could also cause significant loss to company data, as an attacker could delete directories, which may cause a huge loss of data.

Nessus identified an unauthenticated user account vulnerability on the argosoft mail server, however the penetration tester was unable to find technical details for this exploit, and as such just experimented with usage of the add new user function but was unable to find this exact vulnerability. Or, Argosoft may have been configured in such a manner that it was not vulnerable to this exact exploit.

On the log 1 cms vulnerability the penetration tester experimented significantly with the ajax file manager upload feature. The vulnerability was in the data.php folder that was called by the ajax create folder.php, so the penetration tester attempted Injecting code inside the create folder field, however this was unsuccessful. Ultimately, the penetration tester was unsure why this exploit was unsuccessful despite being correctly configured. At the time of this exploit attempt, the penetration tester had not turned off Windows Defender on the target, so that may have detected the exploit and blocked it, however the error on the exploit was no response, so this was probably not the case.

On server 2, exploitation on the BoZon webpage was mostly unsuccessful, however a user was successfully created which is a significant exploit on any webpage. As soon as a user was created, or added via the exploit, it was immediately given superuser permissions on the web page. This allowed the user to change many configurations and permissions, meaning there may have been specific configurations that would allow for further exploitation however the penetration tester did not find a configuration that allowed this. When the exploit was used to execute phpinfo command upon refresh, the server crashed resetting all configurations and webpages.

Both of the HFS web pages allowed exploitation successfully, without either of the servers crashing. When the user attempted to add persistence, the server crashed. This could be because it conflicts with other services running on startup, or a permissions issue. This was attempted multiple times and both servers crashed on every attempt at this.

Once inside the machine, the penetration tester was able to access all information on the network with system permissions, which was expected, and with administrator permissions, showing the network did not follow the recommended policy of least privilege.

DLL hijacking vulnerabilities were also discovered by Winpeas, however the attacker had trouble using the reg command in PowerShell with their user privileges to change the path variable to point towards a new DLL. If more time was had during the Penetration test the Penetration tester may have found a successful vulnerability with this.

Throughout this penetration test vulnerabilities were discovered that the penetration tester was unable to exploit, which may have been set up this way deliberately in order to protect the systems.

## Countermeasures

On initial exploration of the system by the attacker, using the test account, it was discovered that PowerShell was enabled for all user accounts. The penetration tester would recommend disabling PowerShell on all accounts unless needed, for example, by an IT team. This would also be recommended with the ability to run executable files. It would also be recommended that only administrators have permissions to add computers to the Active Directory.

Argosofts latest version is 2.0.4.0 and the version that was running on the network was 1.8.2.9(argosoft.com, 2022). When the latest version was searched for no vulnerabilities were found.

Log 1 cms was running version 2.0, of which the latest version is 2.1 which released in 2011 (sourceforge.net, 2011), however vulnerabilities were found that also affect this, so the penetration tester would recommend switching to a different file hosting and display website that is kept more up to date and is compatible with more modern versions of PHP.

It was not known precisely what version of HFS was running on the server. However, it was still vulnerable to exploitation, so it must not have been the most recent version, which is 2.3m (rejetto.com, 2020). On display on the main HFS page it simply said version 2.3, without a letter at the end denoting the precise version.

Bozon was running version 2.4, which is the latest version, which, however, is still vulnerable to exploitation (github.com/broncowdd, 2016). The failure of the Penetration tester to get full code execution on the server via this exploit does not mean that the application is secure, rather confirms that with that exact set up, at that time the exploitation was not completely successful.

Home FTP Server was used to successfully perform directory traversal and allowed anonymous login. Essentially giving an attacker full system privileges to explore the file system and read files at their leisure, even without an account. To fix this, it would be recommended to use a more up to date FTP server.

For all services running on the network, the penetration tester would highly recommend not giving all of them system privileges, to prevent an attacker having full control over the network in the case of a successful exploit.

The Penetration tester would also recommend implementing a better password policy, enforcing a longer password, at least 14 characters, which must contain numbers, a special character and capital letters. Also, the penetration tester would recommend creating an account lockout, to prevent brute forcing attacks, and both of these recommendations would prevent use of older password leaks being used in a dictionary attack where few passwords would contain 14 or over characters. Or a password manager would be recommended.

The version of PHP that the web applications ran on was severely outdated, being 5.6.30 with the most modern version being 8.3.

The Penetration tester was also made aware that the test account they were given was with a weak password and would recommend that even when creating a temporary account a strong password should be used, in case a malicious actor gained access to the test account.

The penetration tester would also recommend implementing the principle of least privilege. This essentially means to have all your files segregated, and only letting the users access files that they need, such as HR only being able to access HR shares. This would prevent an attacker with an account accessing all systems.

## Future Work

If the Penetration tester was able to have more time, it may have been possible to exploit the vulnerabilities that they were unable to exploit. If they had admin access to the server, and information on the exact services running, they may have been able to reconfigure the server to not crash upon exploitation.

The penetration tester would also run Powersploit and PowerShell empire on the target to search for further pathways for exploitation. The penetration tester would also further explore the XSS vulnerability on Argosoft. All other vulnerabilities discovered by searchsploit would also be tested, and manual hacking instead of the Metasploit modules would be used, and modifications made to the exploits in order to see if the penetration tester could prevent the servers crashing.

# References

www.synopsys.com. (2023). *What is Penetration Testing and How Does It Work? | Synopsys*. [online] Available at: <https://www.synopsys.com/glossary/what-is-penetration-testing.html>. [Accessed 16 December 2023]

O’driscoll, A. (2021). *UK Cyber Security and Cyber Crime Statistics in 2021*. [online] Comparitech. Available at: <https://www.comparitech.com/blog/information-security/uk-cyber-security-statistics/>. [Accessed 16 December 2023]

Howarth, J. (2023). *Alarming Average Screen Time Statistics (2023)*. [online] Exploding Topics. Available at: <https://explodingtopics.com/blog/screen-time-stats>. [Accessed 16 December 2023]

‌Metasploit (2014). *Rejetto HTTP File Server (HFS) - Remote Command Execution (Metasploit)*. [online] Exploit Database. Available at: <https://www.exploit-db.com/exploits/34926>. [Accessed 16 December 2023]

‌

nmap.org. (n.d.). *Preface | Nmap Network Scanning*. [online] Available at: <https://nmap.org/book/preface.html#preface-intro>. [Accessed 16 December 2023]

Vry4n\_ (2021). *WinPEAS - Windows Enum*. [online] VK9 Security. Available at: <https://vk9-sec.com/winpeas-windows-enum/>. [Accessed 16 December 2023]

‌

GitHub. (n.d.). *Dec0ne - Overview*. [online] Available at: <https://github.com/Dec0ne?tab=overview&from=2022-07-01&to=2022-07-31> [Accessed 16 Dec. 2023].

‌sourceforge.net. (n.d.). *log1 CMS - Browse Files at SourceForge.net*. [online] Available at: https://sourceforge.net/projects/log1cms/files/ [Accessed 16 Dec. 2023].

‌

GeeksforGeeks. (2021). *Gobuster - Penetration Testing Tools in Kali Tools*. [online] Available at: <https://www.geeksforgeeks.org/gobuster-penetration-testing-tools-in-kali-tools/>. [Accessed 16 Dec. 2023].

www.argosoft.com. (n.d.). *List of Changes for Mail Server - ArGo Software Design*. [online] Available at: <https://www.argosoft.com/rootpages/mailserver/ChangeList> [Accessed 16 Dec. 2023].

‌

Bronco (2023). *BoZoN*. [online] GitHub. Available at: https://github.com/broncowdd/BoZoN [Accessed 16 Dec. 2023].

‌www.rejetto.com. (n.d.). *HFS ~ HTTP File Server*. [online] Available at: <https://www.rejetto.com/hfs/?f=dl> [Accessed 16 Dec. 2023].

‌

HackTricks. (2020). *File Upload*. [online] Available at: <https://book.hacktricks.xyz/pentesting-web/file-upload>. [Accessed 16 Dec. 2023].

‌

‌

# Appendices

Note that Appendices should be referenced in the main body of the text.

## Appendix A

**Server 1 nmap scan**

A screenshot of a computer

Description automatically generated

**Server 1 with -A option**

A black dragon with white text

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screen shot of a computer program

Description automatically generated

A computer screen shot of a computer

Description automatically generated

**Server 2 nmap scans**

Scan using syntax in figure 2

**A screen shot of a computer

Description automatically generated**

**Server 2 with -A flag.**

**A computer screen shot of white text

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**A computer screen shot of a computer

Description automatically generated**

**A computer screen shot of a computer program

Description automatically generated**

## Appendix B

Readme.txt on 192.168.10.1:90/readme.txt.

A screenshot of a computer

Description automatically generated