

DLL Injection and OllyDBG

CSC 471 - 02

Tyler Prehl

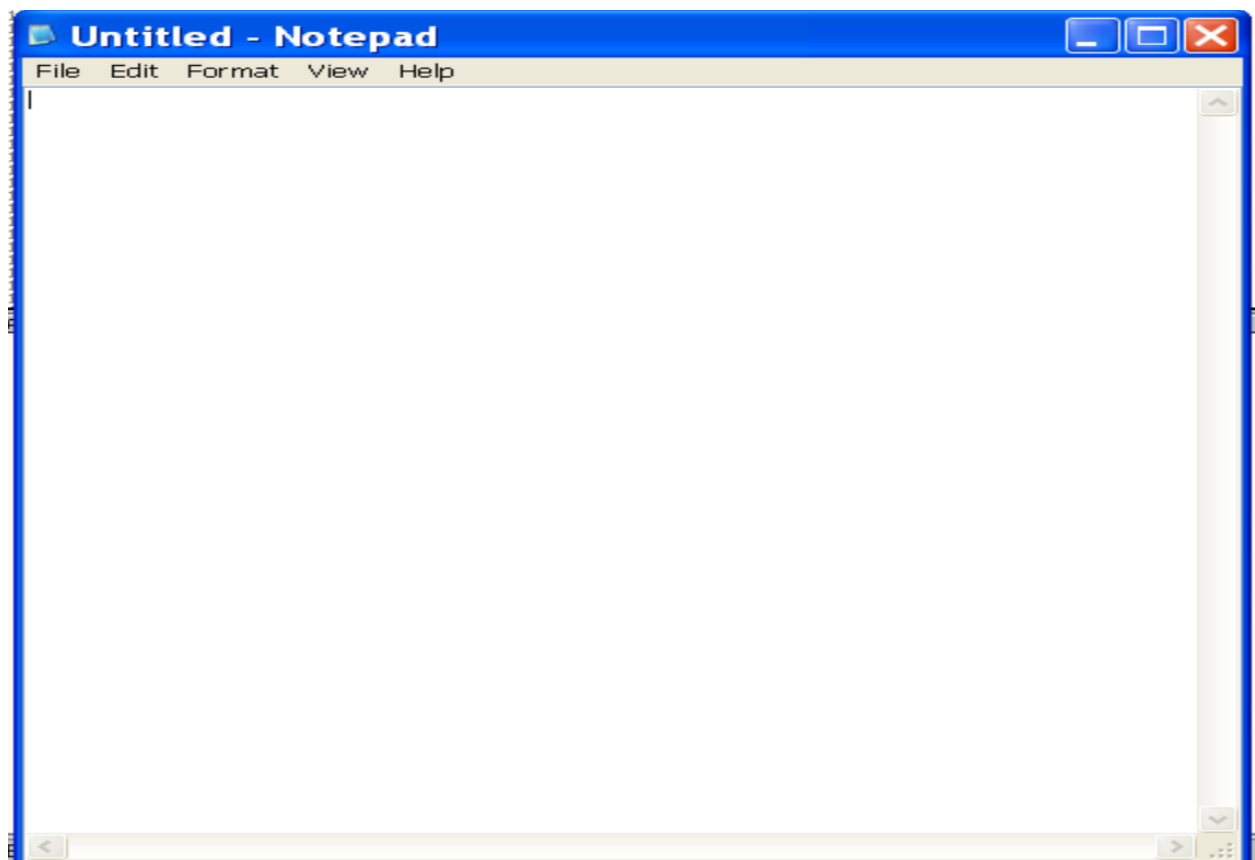
2/12/2022

## Introduction

The purpose of this lab is to provide a better understanding of a couple things. First and foremost, it will provide insight into how to execute a dll file injection attack. Secondly, it provides background knowledge on how to navigate various applications, such as VirtualBox, Process Explorer, and OllyDbg. Most importantly, it provides exposure to reading and editing the hex values in memory to recreate files the way we want them from executable files.

## Analysis and Results

To start the dll file injection attack, we need a process to attack, so I opened Notepad.



Next, I needed to know what the process ID of the Notepad application was so that the InjectDll.exe file could have a target application to inject with a new dll file.



I then navigated to 10010B20 in memory, because it was the actual address in memory where the string that gets printed to DebugView is stored.

10010B1F JNE SHORT myhack.10010B20  
 10010B21 PUSH myhack.10010B20  
 10010B22 CALL DWORD PTR DS:[4\*KEBNEI+2],OutputDebugStringA

String = "<myhack.dll> Injection!!!  
 myhack.dll

Address	Hex dump	ASCII
10010B20	3C 00 6D 00 79 00 68 00 61 00 63 00 68 00 2E 00	<.m.y.h.a.c.k..
10010B30	64 00 6C 00 6C 00 3E 00 20 00 49 00 6E 00 6A 00	d.l.l.>.I.n.j.
10010B40	65 00 63 00 74 00 69 00 6F 00 6E 00 21 00 21 00	e.o.t.i.o.n.t.t.
10010B50	21 00 20 00 2D 00 2D 00 20 00 43 00 53 00 43 00	!.-.-.C.s.C.
10010B60	20 00 34 00 39 00 37 00 2F 00 35 00 30 00 33 00	.4.9.7./5.0.3.
10010B70	20 00 2D 00 2E 00 20 00 53 00 69 00 20 00 43 00	.-.-.S.l..C.
10010B80	68 00 65 00 6E 00 00 00 00 00 00 00 00 00	h.e.n.....
10010B90	00 00 00 00 00 51 49 5C 00 00 00 00 00 00	...iDI\...@...

With direct access to the location in memory, I edited the binary to include “Hello World! - Tyler Prehl” and copied it to an executable file to be saved as a new dll - myhack\_Prehl.dll.

myhack.<ModuleEntryPoint>

Address	Hex dump	ASCII
10010B20	3C 00 6D 00 79 00 68 00 61 00 63 00 68 00 2E 00	<.m.y.h.a.c.k..
10010B30	64 00 6C 00 6C 00 3E 00 20 00 48 00 65 00 6C 00	d.l.l.>.H.e.l.l.o..
10010B40	6C 00 6F 00 20 00 57 00 6F 00 72 00 6C 00 64 00	l.o.w.o.r.l.d.
10010B50	21 00 20 00 2D 00 20 00 54 00 79 00 6C 00 65 00	!.-.-.T.y.l.e.
10010B60	72 00 20 00 50 00 72 00 65 00 68 00 6C 00 20 00	r..P.r.e.h.l.
10010B70	20 00 20 00 20 00 20 00 20 00 20 00 20 00	.....
10010B80	20 00 20 00 20 00 00 00 00 00 00 00 00 00	.....
10010B90	00 00 00 00 AD 51 49 5C 00 00 00 00 00 00	...iDI\...@...
10010BA0	00 00 00 00 AD 0C 01 00 AD FE 00 00 00 00 00	C...a.0.a...@...
10010BB0	00 51 49 5C 00 00 00 00 00 00 00 00 14 00 00	iDI\...@...@...

After some failed attempts in trying to edit the binary, I finally was able to properly save and run myhack\_Prehl.dll:

```

2  343.03531594 [1460] <myhack.dll> Hello World! - Tyler Prehl --
3  560.43798828 [796] <myhack.dll> Hello World! - Tyler Prehl

```

## Discussion and Conclusion

With a successful dll injection, this lab was a success. I achieved all of my goals, including learning how to properly use VirtualBox and OllyDbg, as well as learning a bit about how to navigate through memory and edit certain pieces to change the outcome of an executable file.

This will be useful going forward as our class dives into attacks that require a strong understanding of how to use OllyDbg, and also for any future dll-related attacks we may attempt.

