What You Need

The Windows Server 2008 VM handed out by your instructor. This project can be done on other

machines, including Windows 10, but first you'll need to install Windbg and LiveKD

Purpose

Practice using simple WinDbg commands. We'll use LiveKd, a utility that makes some limited kernel

debugging possible with a single computer. LiveKd is read-only -- you can look at kernel processes

and data structures, but cannot modify a running system or use breakpoints.

Using LiveKd

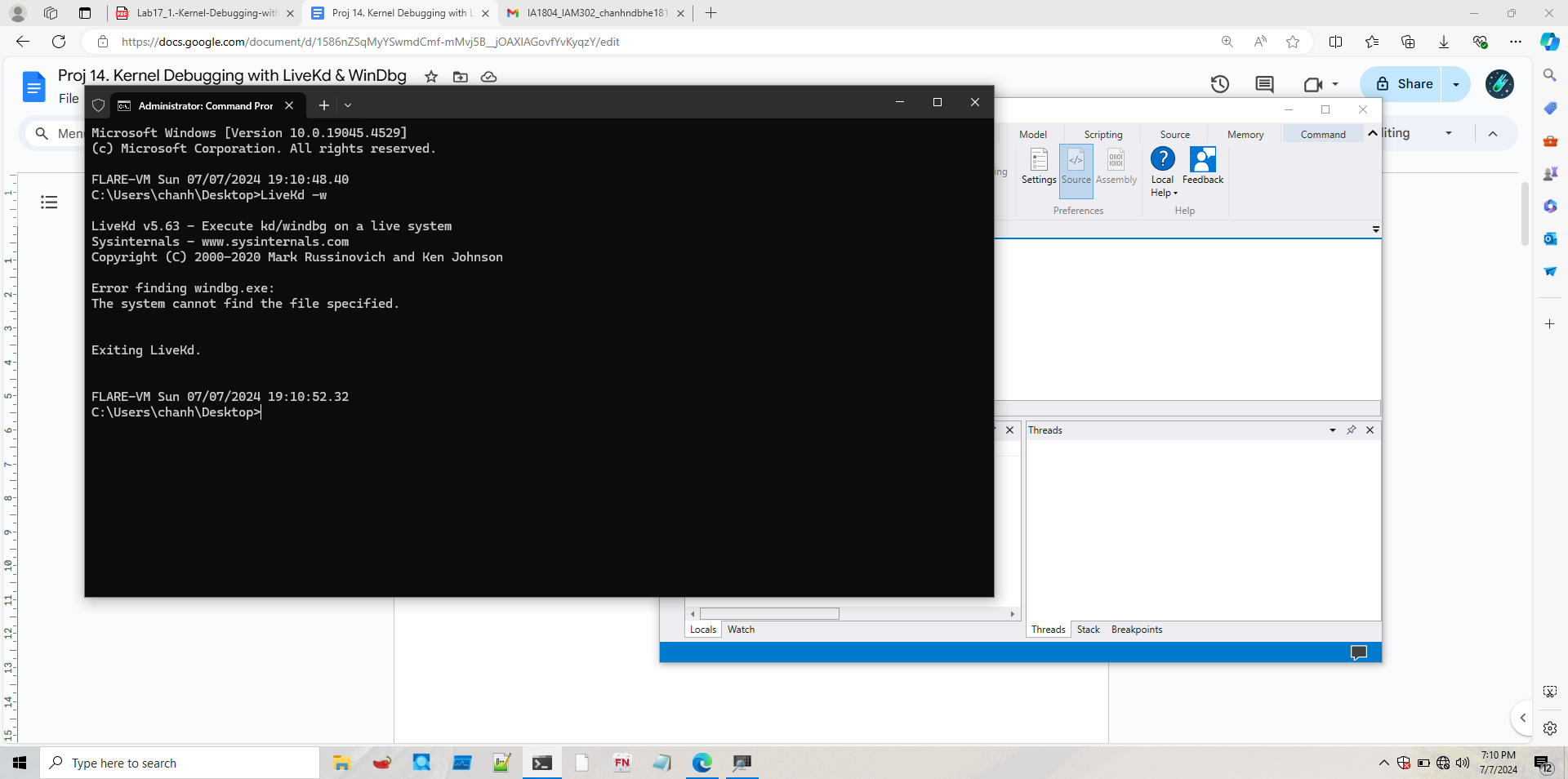
In a Command Prompt window, execute this command:

livekd -w

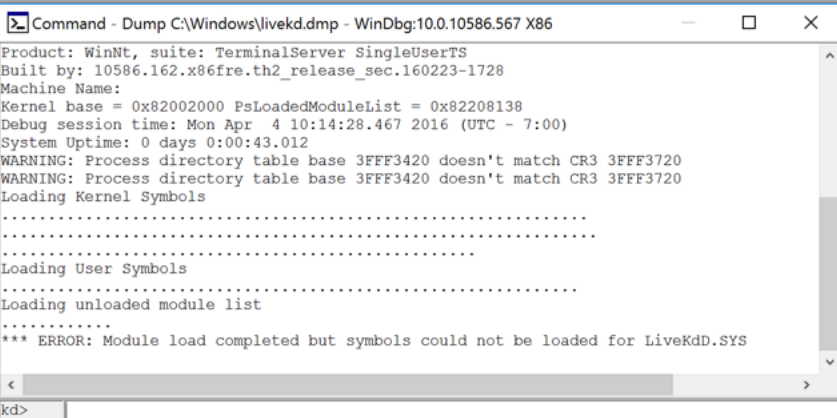
When Livekd starts, it asks you whether to set the \_NT\_SYMBOL\_PATH automatically, as shown

below.

Type y and press Enter



Livekd asks "Enter the folder to which symbols download". Press Enter to accept the default option. Windbg launches, as shown below



This is a strange combination of a GUI and command-line, like the other debuggers we've used.

Commands are typed into the box at the bottom and the results appear in the large top pane.

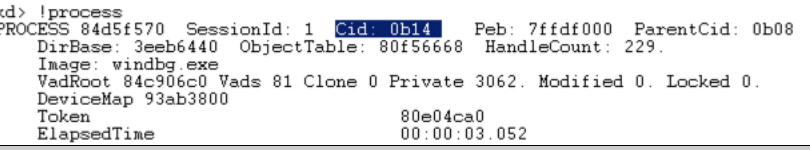
At the bottom of the Command window, in the command bar, execute this command:

!process

You should see the "kd> !process" command, and its output, showing information about the windbg

process, including its Cid number, as shown below.

When I did it, the Cid was 0b14 in hexadecimal, which is 11\*256 + 16 + 4 = 2836

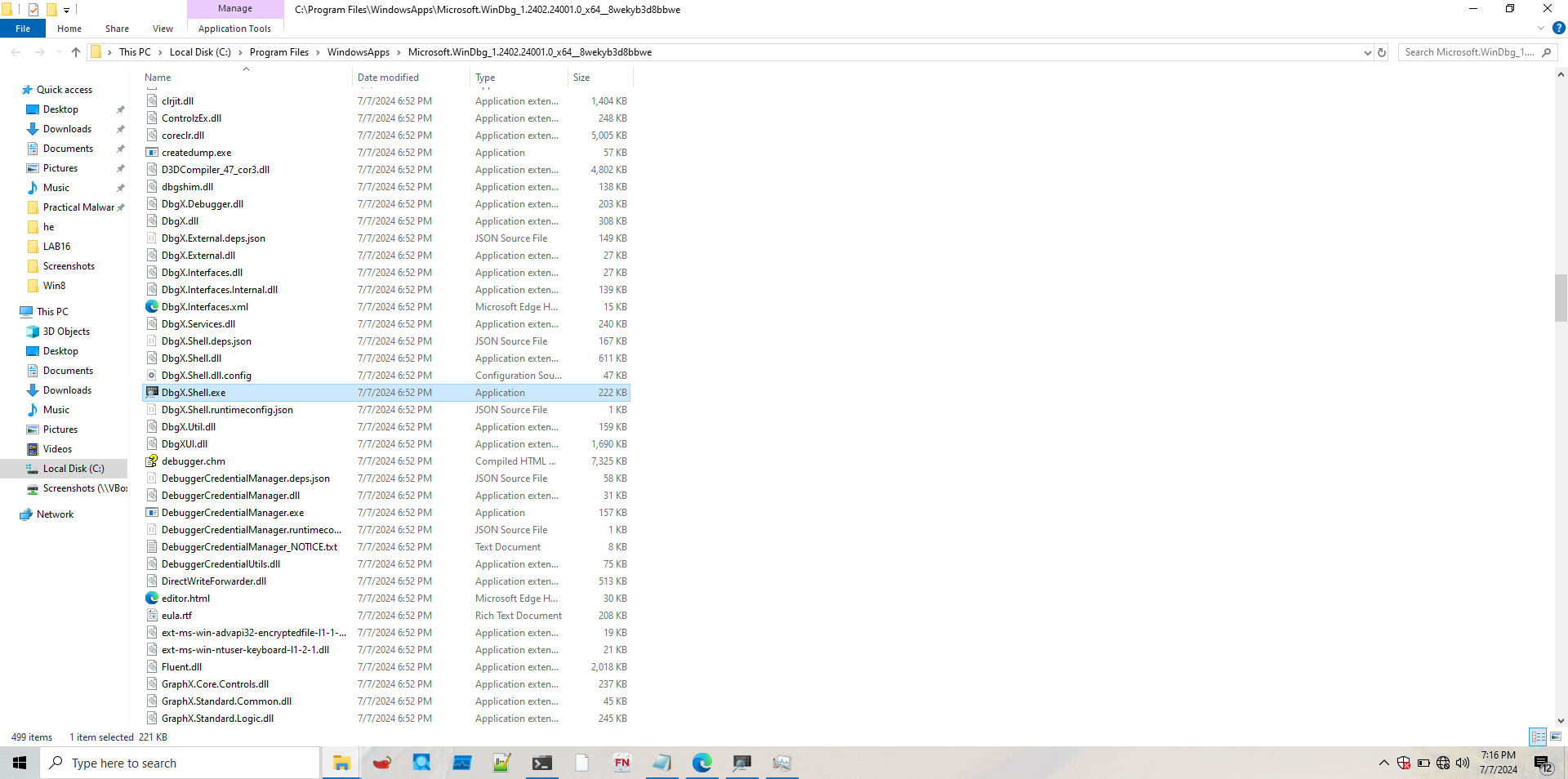


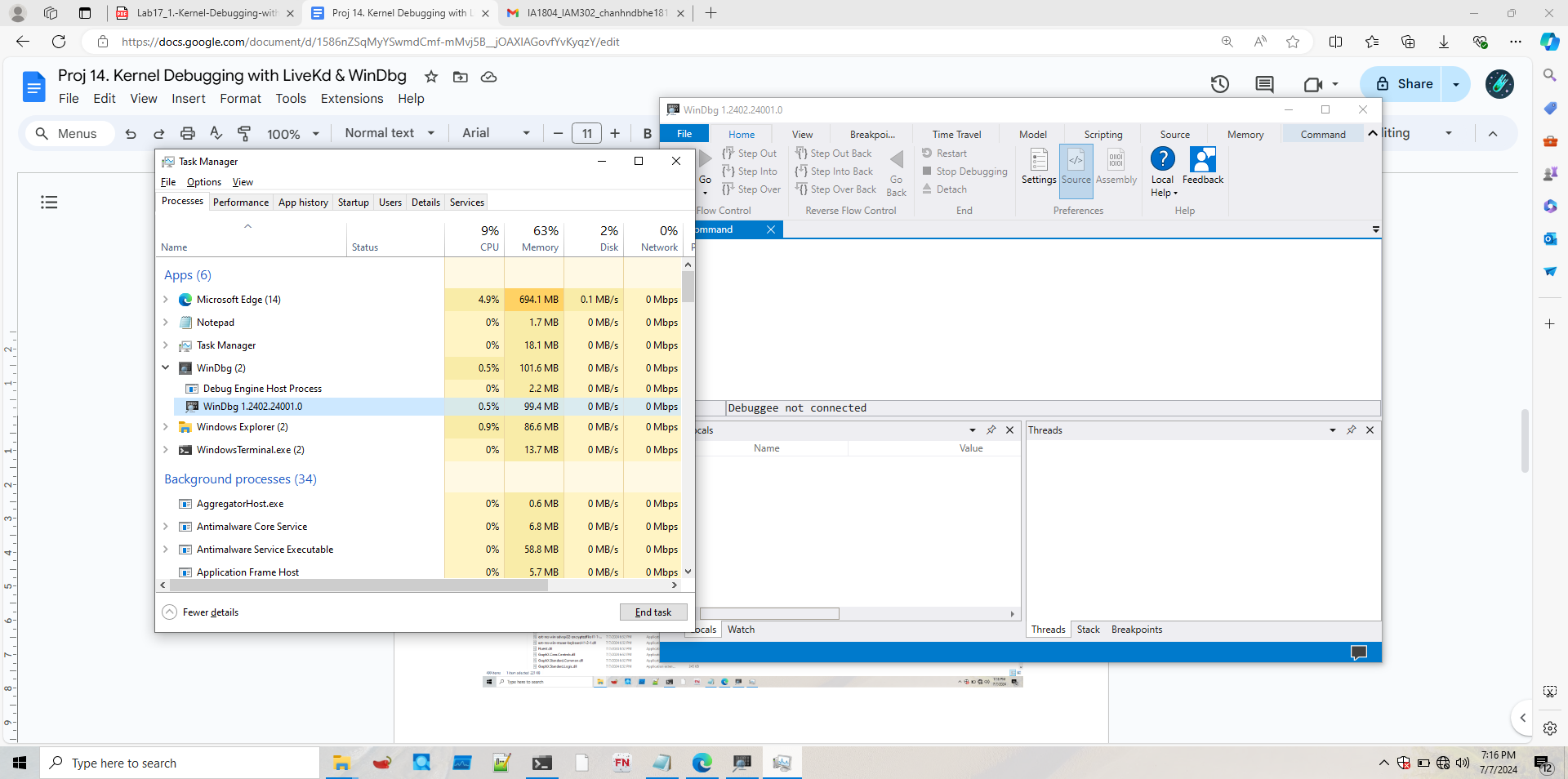
Viewing Processes with Task Manager

At the bottom of the desktop, point to an unused portion of the taskbar and right-click. Click "Task

Manager". In Task Manager, click the Processes tab.

Find the windbg process, and its PID, as shown below. It should match the Cid from Windbg



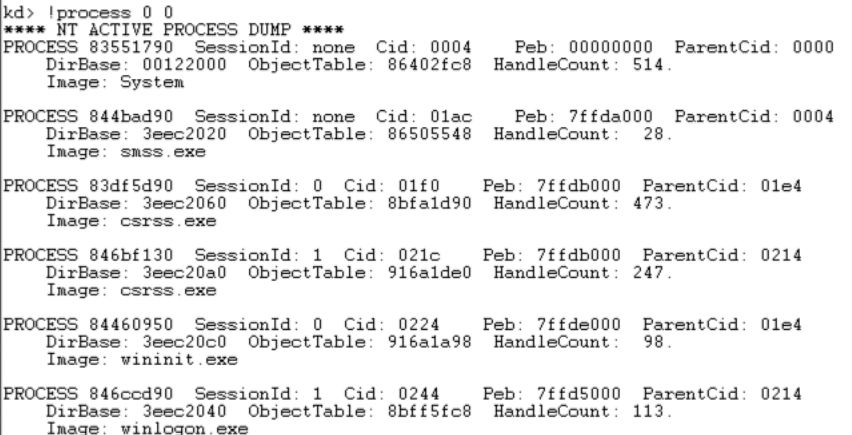


Close Task Manager. In Windbg, at the bottom of the Command window, in the command bar, execute

this command:

!process 0 0

You see a long list of all processes, as shown below

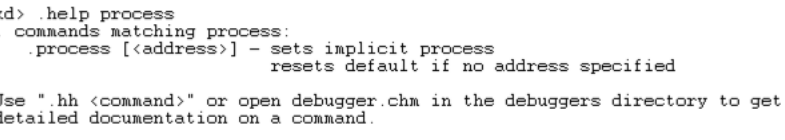


Online Help

At the bottom of the Command window, in the command bar, execute this command:

.help process

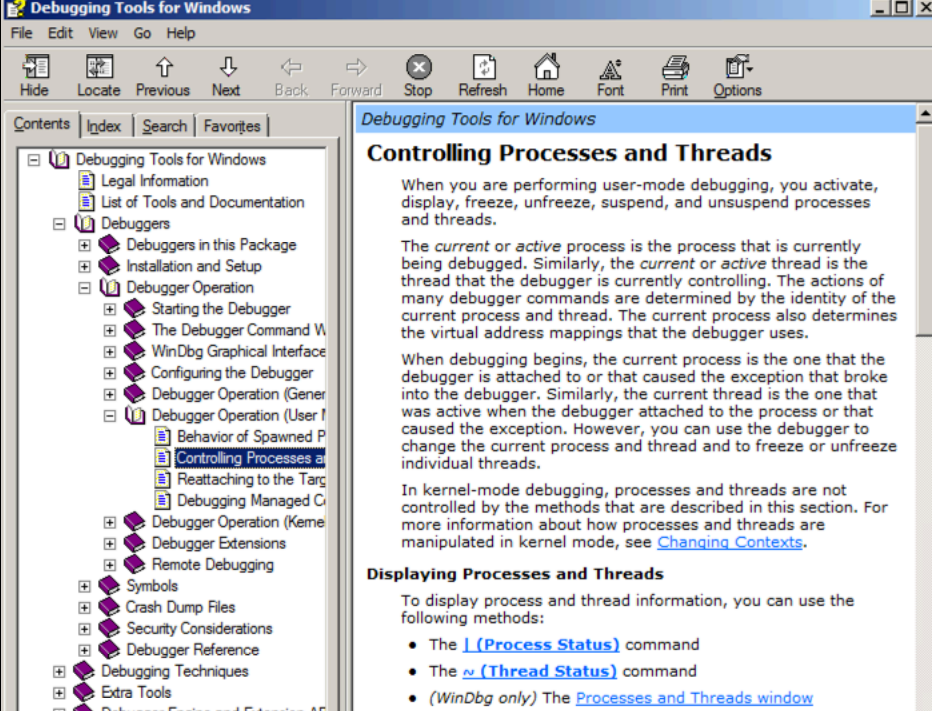
You see a brief help message about the "process" command, as shown below



At the bottom of the Command window, in the command bar, execute this command:

.hh process

You see a much more complete help window, as shown below



Listing Modules with lm

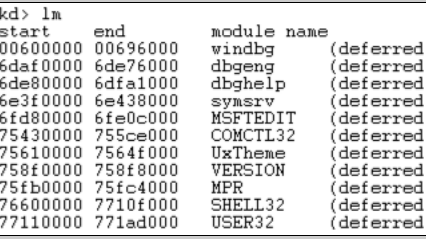
At the bottom of the Command window, in the command bar, execute this command:

lm

A long list of all loaded modules scrolls by.

Scroll back to see the lm command you entered, and the first few loaded kernel modules, as shown

below.



Viewing Memory

In WinDbg, execute this command:

dd nt

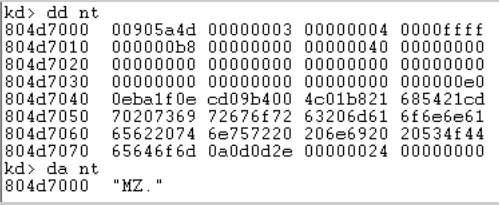
You see the first several bytes of Ntoskrnl.exe, as shown below.

This may be more familiar in ASCII.

In WinDbg, execute this command:

da nt

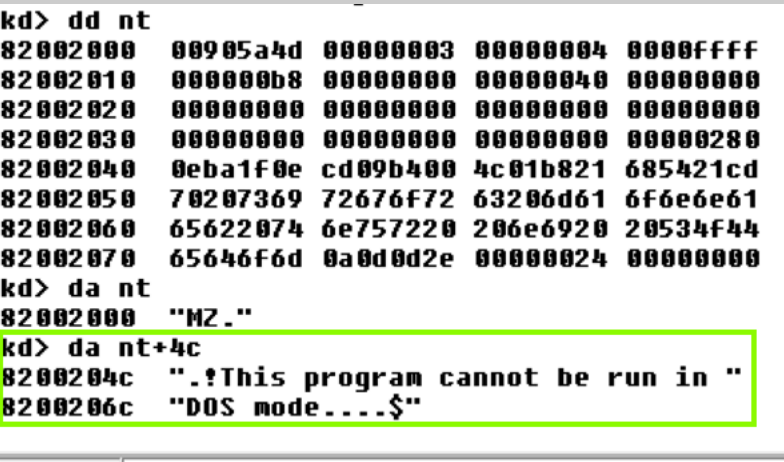
You see the characters "MZ" --they are at the start of every EXE file



In WinDbg, execute this command:

da nt+4c

You see the message "This program cannot be run in DOS mode", as shown below



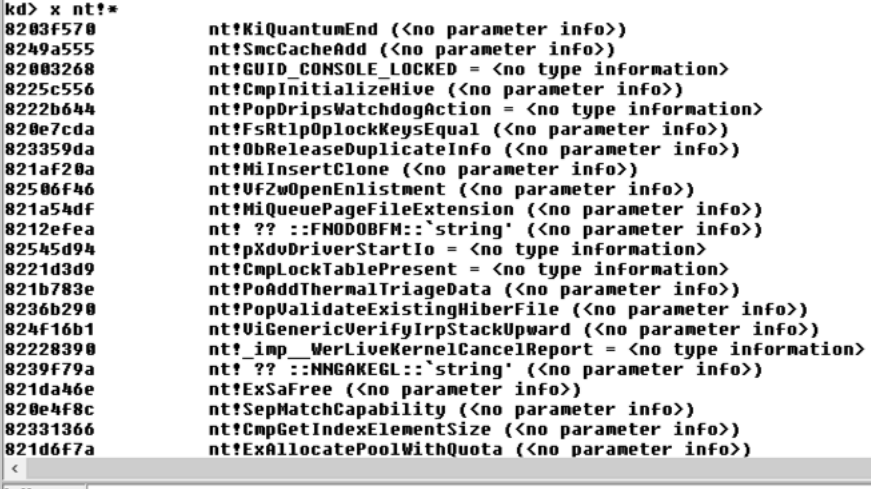
Searching for Functions

In WinDbg, execute this command:

x nt!\*

This finds all the functions in Ntoskrnl.

There are a lot of them, as shown below



In WinDbg, execute this command:

x nt!\*Create\*

This finds all the functions in Ntoskrnl that contain the word "Create".

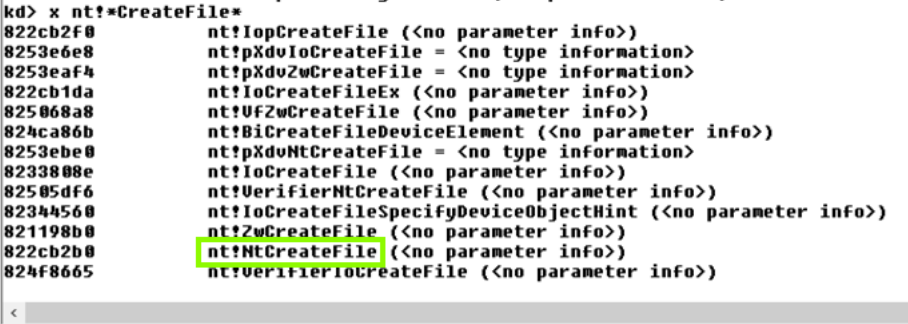
There are a lot of them, too.

In WinDbg, execute this command:

x nt!\*CreateFile\*

This finds all the functions in Ntoskrnl that contain the word "CreateFile".

There are only about ten of those, including "nt!NtCreateFile", as shown below

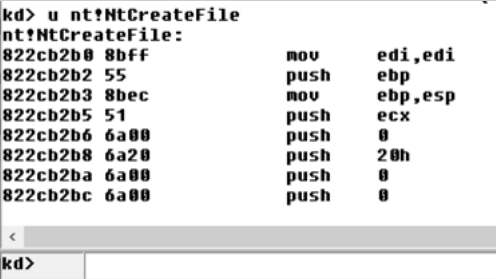


Unassembling a Function

In WinDbg, execute this command:

u nt!NtCreateFile

This shows the first few bytes of the function, disassembled, as shown below



To see more of this function, it helps to use the WinDbg Disassembly window.

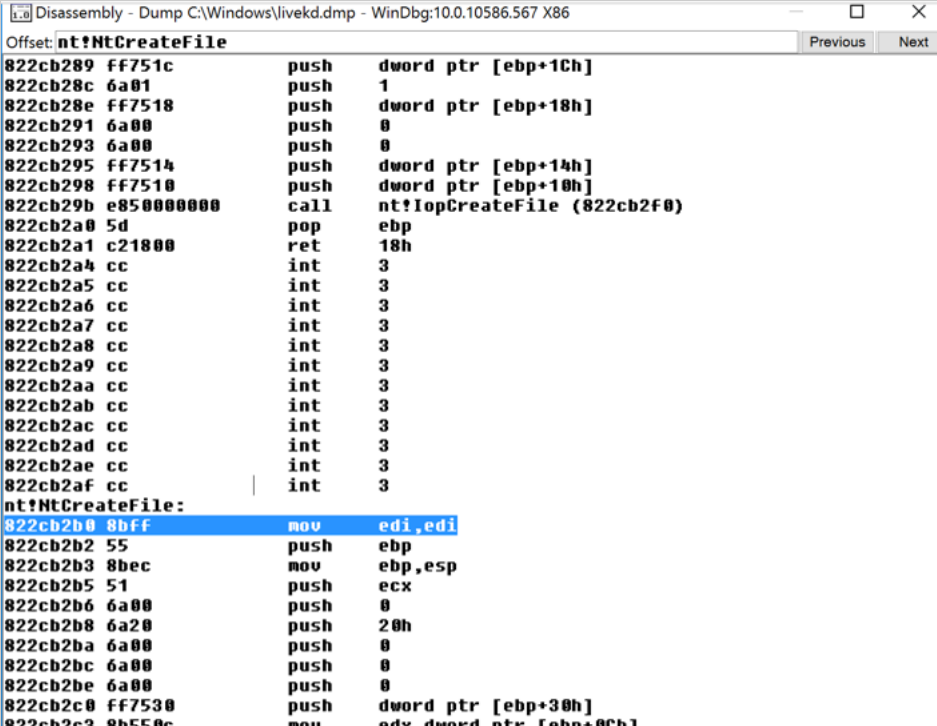
If the Command window is maximized, make it smaller.

From the WinDbg menu bar, click View, Disassembly.

In the Offset bar at the top, enter

nt!NtCreateFile

This shows the assembly code before and after the start of the NtCreateFile function, as shown below



In the Offset bar at the top, enter

nt!NtCreateFile+16

Resize this window to make the entire function visible. Drag the mouse through it to highlight the entire function, as shown below

