Sai Sasaank Srivatsa Pallerla

As you edit this file, type your name here

NO COLLABORATION OF ANY KIND

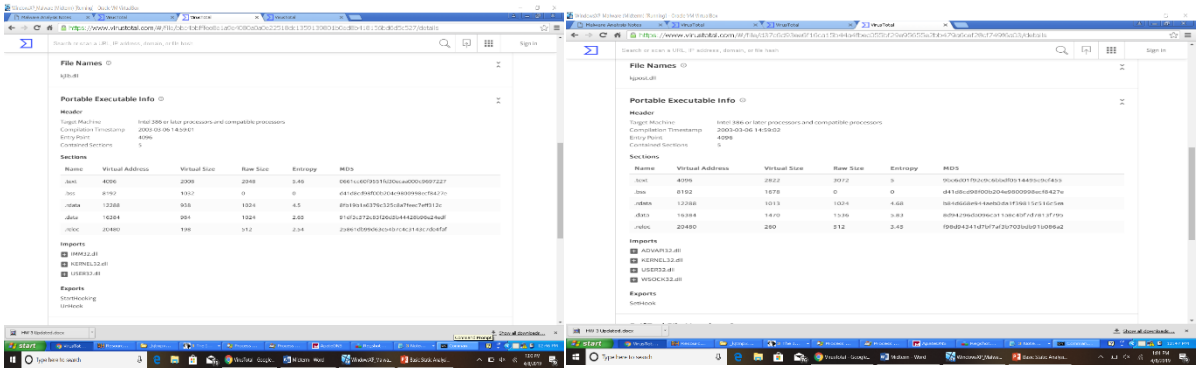
CMSC 491/691 Malware Analysis
Midterm Exam Spring 2019

Extract midterm2019.7z on a Windows XP virtual machine. The password is infected. Use basic static analysis tools to answer the following questions about midterm2019.exe:

1) **Is midterm2019.exe packed? Justify your answer with three reasons. (9 pts)**

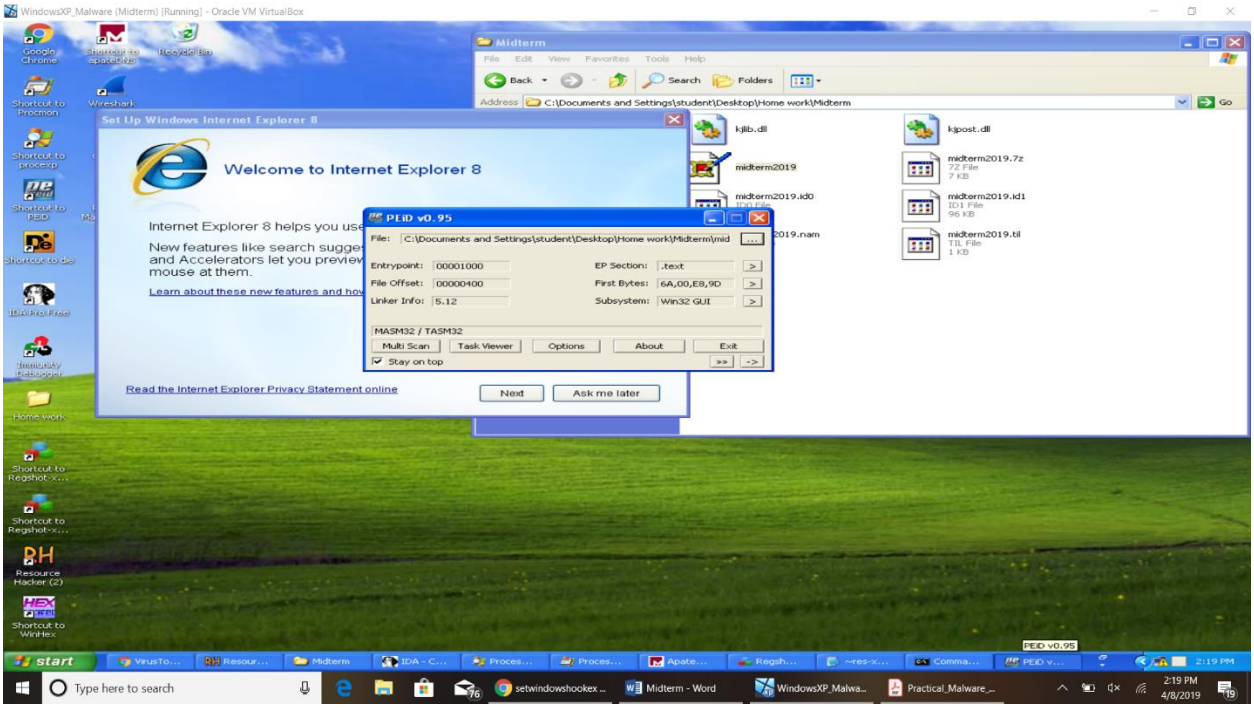
Packed (Y/N): N

a. Entropy : Unusual levels in virtual size is observed. Entropy of all the files in midterm2019 less than 7 but we can notice that the entropy of .bss segments in kjlib.dll and kjpost.dll is 0 but this is not an indicator of packing because it they are the files stored in resource segment.



b. Readable strings: There are a lot of string in midterm2019.exe that are suspicious, these suspicious strings include the strings from kjlib.dll and kjpost.dll.

c. PEid: Nothing came up when the files is uploaded to PEid. It just gave information about the assembly code.



2) List five strings, imports, and/or metadata in midterm2019.exe that you believe are suspicious. Describe why each of them is suspicious. (10 pts)

a. SetWindowsHookEx: This provides the malware an easy way to load the DLL files in the user interface of a system. It sets a hook function, that is when an set event occurs perform this function. Here the malware is a keylogger, so the this helps to monitor the entries that are made using keyboard.

b. CallNextHookEx: This function is gets its control from SetWindowsHookEx, and it is call the next function in the hook chain. Here, probably it calling the next function – which might be string the keyboard entries into a text file.

c. GetKeyboardState: The function gives status of a specfic key on the keyboard. This function is very common among the keyloggers.

d. Badguys.com: This is one of the strings in midterm2019.exe and after further examining the malware we can notice that the text file created by malware is being sent to a website and this being a website there are high chances it is being sent here. The name by itself is suspicious.

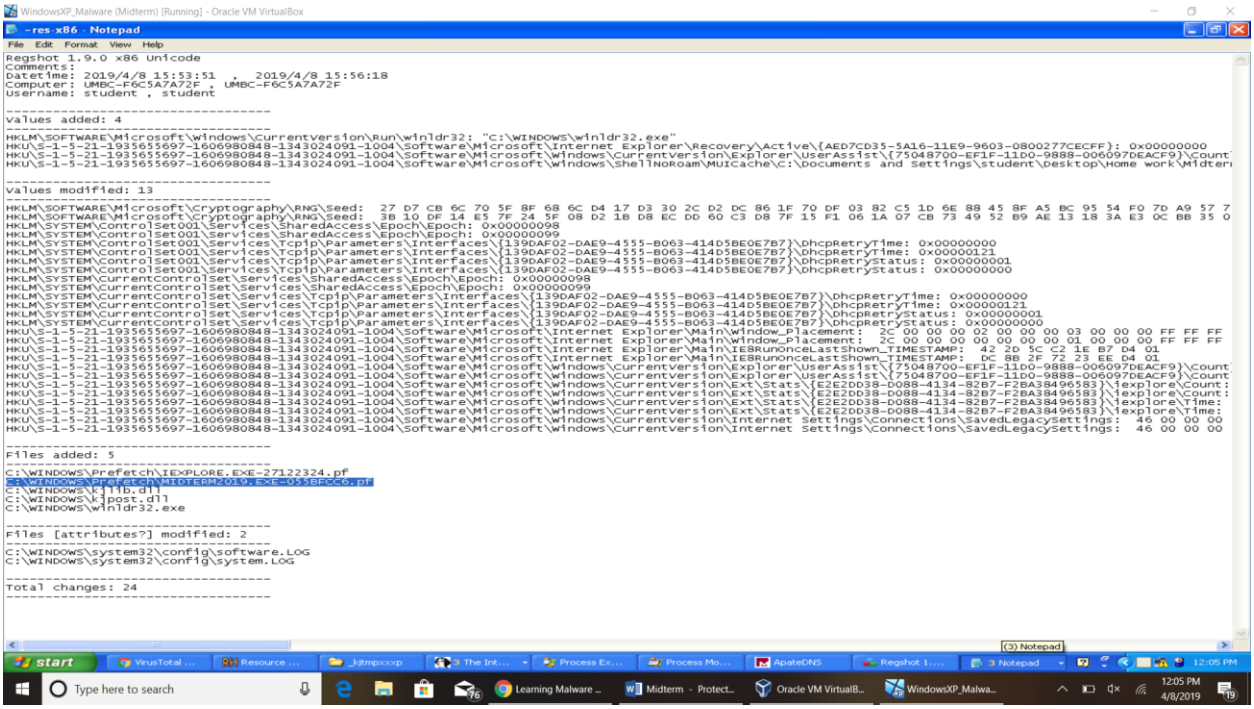
e. Kjpost.dll and kjlib.dll: These are the two DLL files that are hidden in the resources segment. These DLLs are extracted later, during runtime, and stores in System32 folder. These Dlls are responsible for monitoring the keyboard entries, saving them to a text file and sending them a doomain.

f. GetTempPath: Returns the temporary file path. Usually it uses this function if it wants to reads or writes any files in the temporary file path.

Prepare any tools you need to perform basic dynamic analysis and set your VM to **Internal Network**. Take a snapshot of your VM so that you can easily revert to this point later. Run midterm2019.exe and answer the following questions:

3) Where does midterm2019.exe copy itself to? Make sure to list the full file path. (2 pts)

Midterm2019.exe copies itself to: C:\Windows\winldr.exe

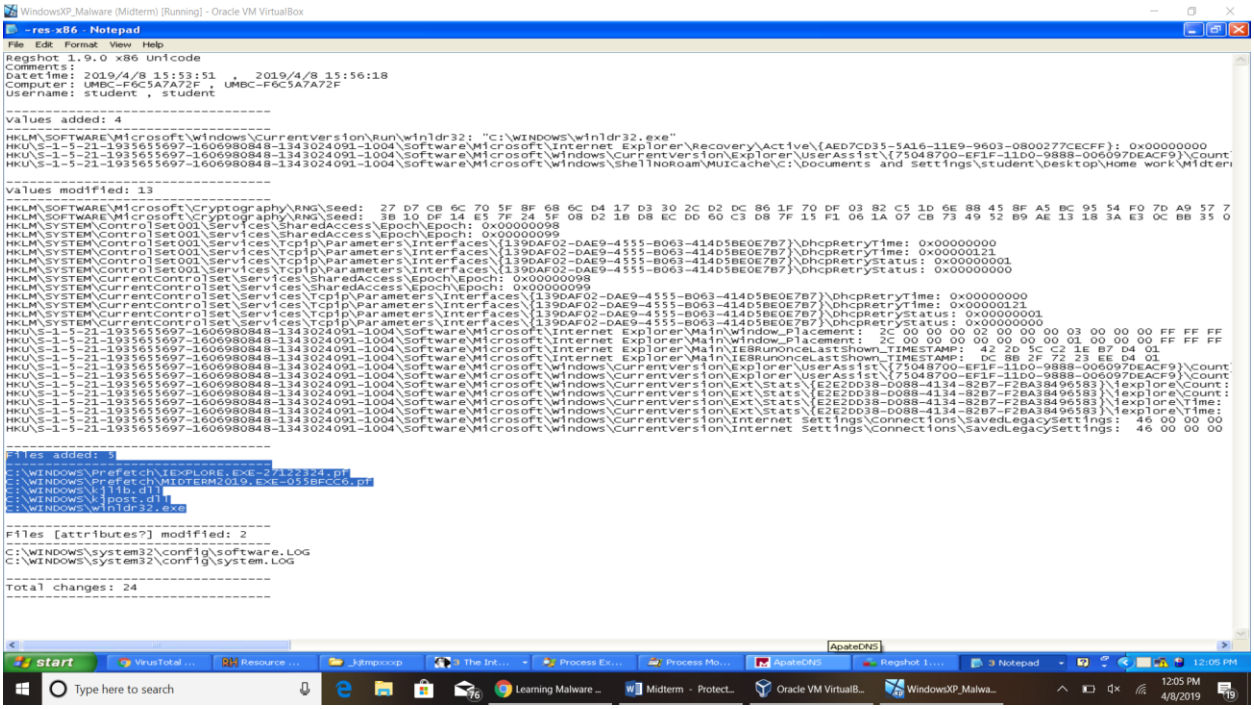


It can also probably copy itself as a pf file Midterm2019.exe copies itself to: C:\Windows\Prefetch\MIDTERM2019.EXE-055BFCC6.pf

4) What other PE files does the malware create, if any? (4 pts)

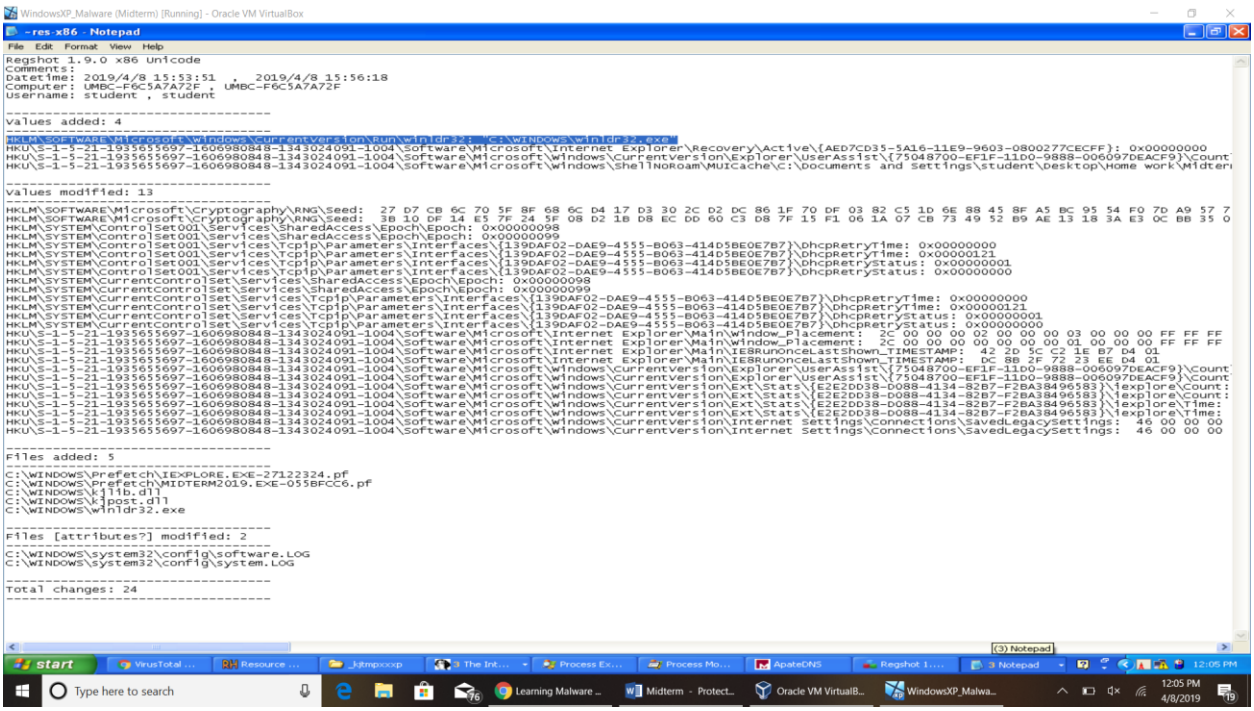
After comparing the two files from regshot we can notice 5 PE files are created

1. IEXPLORE.EXE
2. kjlib.dll
3. kjpost.dll
4. winldr.exe



5) How does the malware gain persistence? (4 pts)

The malware gains persistence by installing itself as a service which goes by the name: winldr.exe. This can be obtained from the Registry key: HKML\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\winldr.exe



Use IDA Pro to answer the following question about midterm2019.exe:

6) **In a sentence or two, summarize what sub_401428 in midterm2019.exe does. (8 pts)**
   This function is responsible for creating two files 1. Kjlib.dll 2. Kjpost.dll. This function first checks if these files already exist and if they don't it will create them. These library files are stored in resource segment with 1001 and 1002 headings. This function takes that BINTYPE and address and creates those two files.

7) **In a few sentences, summarize what sub_4011FA does. How long is the timer set? (10 pts)**
   This function loads the DLL file, kjlib.dll, then calls StartHooing and UnHook, using GetProcAddress, when the event is done it sets a timer for 60000 milliseconds (0EA60h).
   After the function calls StartHooking it copies the Hmodule addres to eax register and later call it in the later stages.

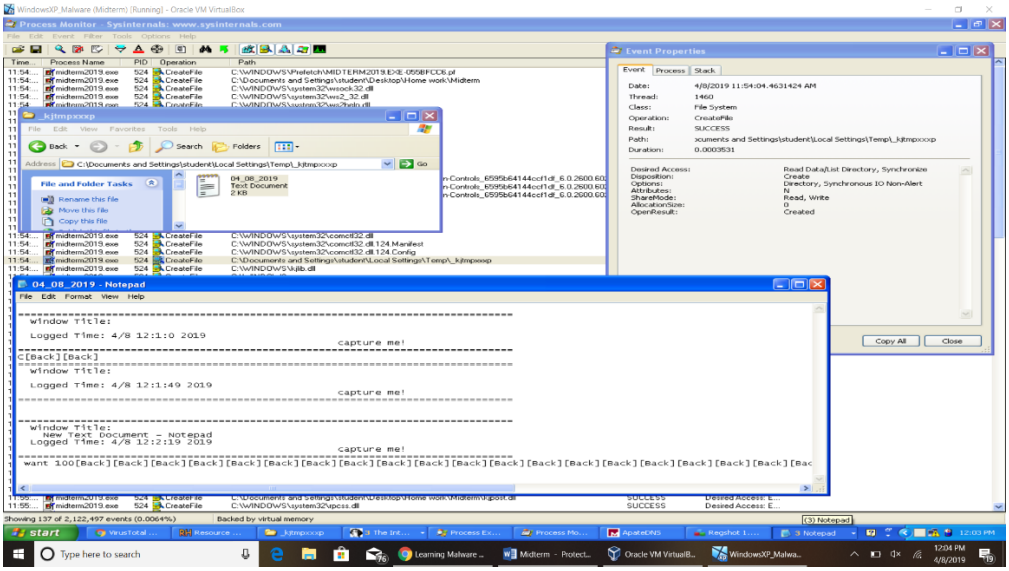
8) **In a few sentences, summarize what sub_4012D9 (StartAddress) does. (8 points)**
   In this function the malware first checks for internet connection and if there is connectivity it will load kjpost.dll, open Internet Explorer, sleeps for sometime, loads the message and file (opens a thread to do this) that wants to send over the internet and then sends it.

9) **Why does the malware call gethostbyname at 0x4012E4? (4 pts)**
   This function uses the WindowsAPI call gethostbyname to connect to google.com. The malware is probably doing this to check if there is internet connection because in the later stages it will try to send a text file.

Regarding the file that midterm2019.exe creates at 0x4014F0:

10) **Provide a detailed analysis (at least one paragraph) of what this file does. Make sure to describe which Windows API functions are used perform any malicious activity. (16 pts)**
The file that the malware creates at 0x4014F0 is: kjlib.dll. This DLL has all the WindowsAPI calls that the malware will use to get the keyboard information (to act as a keylogger). Series of operations that it can help the malware perform are SetHooks, UnHook, Keyboard states and Creates files.
This dll file helps malware create a file and then save whatever is being typed and any event activity.
Windows API functions used to perform malicious activity:
    1. SetWindowsHook: Installs an application-defined hook procedure into a hook chain. Helps monitor the system for certain types of events. These events are associated either with a specific thread or with all threads in the same desktop.
    2. UnhookWindowsHook: Removes a hook procedure installed in a hook chain by the SetWindowsHook function.
    3. GetKeyboardState: Copies the status of the 256 virtual keys to the specified buffer.
    4. CallNextHook: Passes the hook information to the next hook procedure in the current hook chain. It can call the function either before or after processing the hook information.
    5. CreateFile: Used to create a new File with the described file parameters.
    6. WriteFile: Used to write into a file.
    7. GetWindowText: Copies the text of the specified window's title bar (if it has one) into a buffer.

11) **This file creates a .txt file on the infected system. What is the full path of the .txt file? Provide a screenshot of its contents. (9 pts)**
Full path of _kjtmpxxxp is:
C:\Documents and Settings\student\Local Settings\Temp_kjtmpxxxp



Regarding the file that midterm2019.exe creates at 0x4015D8:

12) **Provide a detailed analysis (at least one paragraph) of what this file does. Make sure to describe which Windows API functions are used perform any malicious activity. (16 pts)**
The file that the malware creates at 0x4015D8 is: kjpost.dll. This DLL has all the WindowsAPI calls that the malware will use to send the text file it created (_kjtmpxxxp) to a specific domain (or probably a mail address).
Using this Dll malware checks for internet connection and if there is connectivity it will load kjpost.dll, loads the message and file (opens a thread to do this) that wants to send over the internet and then sends it.
Windows API functions used to perform malicious activity:
    1. Socket: Creates sockets which is bound to specific service provider.
    2. Send: Sends data on a connected socket.
    3. Revc: Receives data from a remote machine. Malware often uses this function to receive data from a remote command-and-control server.
    4. WSAStartup: Used to initialize low-level network functionality probably to connect to something. Maybe open a socket.
    5. Gethostbyname: Used to perform a DNS lookup on a hostname prior to making an IP connection to a remote host.
    6. GetTempPath: Returns the temporary file path. Usually it uses this function if it wants to reads or writes any files in the temporary file path.
    7. All the hook functions described in 10)