Song Yang (sy540)
Xin Yang (xy213)
Zhuohang Li (zl299)

Homework6

Lab07-01.exe:

1.It is creating a service “MalService” to the start table.

```
sub     esp, 10h
lea     eax, [esp+10h+ServiceStartTable]
mov     [esp+10h+ServiceStartTable.lpServiceName], offset aMalservice ; "MalService"
push    eax             ; lpServiceStartTable
mov     [esp+14h+ServiceStartTable.lpServiceProc], offset sub_401040
mov     [esp+14h+var_8], 0
mov     [esp+14h+var_4], 0
call    ds:StartServiceCtrlDispatcherA
push    0
push    0
call    sub_401040
add     esp, 18h
retn
_main endp
```

2. As is shown in the screenshot, it checks for mutex HGL345 on call of sub_401040. If the mutex is find, the program will exit, otherwise it will create a mutex with HGL345. So we can infer that it is using mutex to avoid running this program multiple times simultaneously.

```
SystemTime= SYSTEMTIME ptr -400h
FileTime= _FILETIME ptr -3F0h
Filename= byte ptr -3E8h

sub     esp, 400h
push    offset Name     ; "HGL345"
push    0               ; bInheritHandle
push    1F0001h         ; dwDesiredAccess
call    ds:OpenMutexA
test    eax, eax
jz      short loc_401064
```

```
push    0               ; uExitCode
call    ds:ExitProcess
```

```
loc_401064:
push    esi
push    offset Name     ; "HGL345"
push    0               ; bInitialOwner
push    0               ; lpMutexAttributes
call    ds:CreateMutexA
push    3               ; dwDesiredAccess
push    0               ; lpDatabaseName
push    0               ; lpMachineName
call    ds:OpenSCManagerA
```

3. Look for service “MalService” and mutex “HGL345”.

4. As is shown it is using IE8.0 to connect to “http://www.malwareanalysisbook.com”

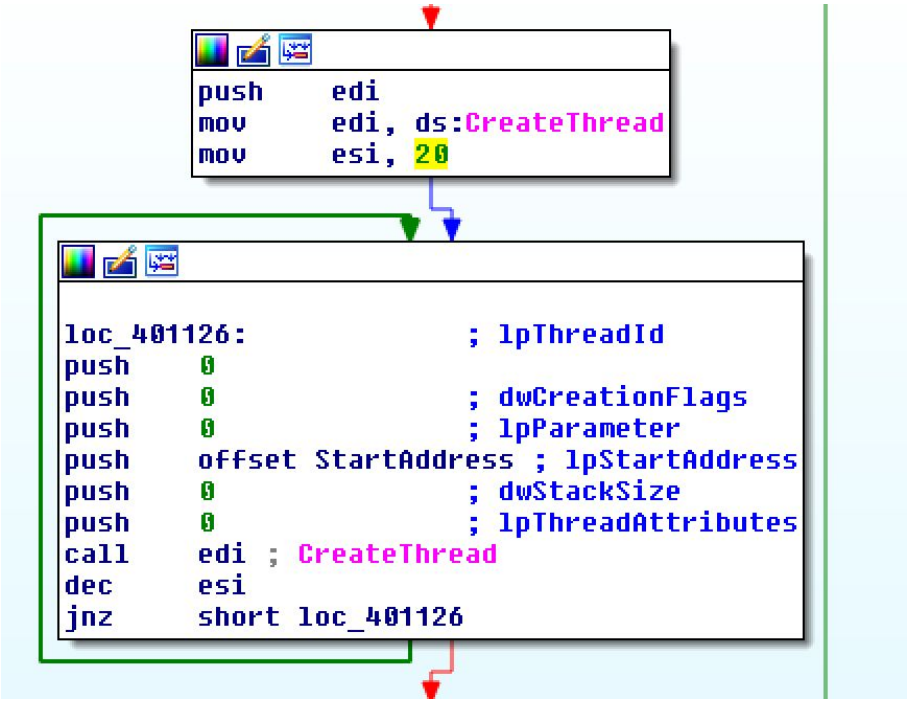
```
lpThreadParameter= dword ptr  4

push    esi
push    edi
push    0               ; dwFlags
push    0               ; lpszProxyBypass
push    0               ; lpszProxy
push    1               ; dwAccessType
push    offset szAgent  ; "Internet Explorer 8.0"
call    ds:InternetOpenA
mov     edi, ds:InternetOpenUrlA
mov     esi, eax
```

```
loc_40116D:             ; dwContext
push    0
push    80000000h       ; dwFlags
push    0               ; dwHeadersLength
push    0               ; lpszHeaders
push    offset szUrl    ; "http://www.malwareanalysisbook.com"
push    esi             ; hInternet
call    edi ; InternetOpenUrlA
jmp     short loc_40116D
StartAddress endp
```

5. As we can see from the following section, the edx is initiated with 0, and all system time variables are assigned with value in edx. Latter on year is assigned with 2100. Then call SetwaitableTimer and WaitForSingleObject to wait till that time.

```
push    esi             ; hSCManager
call    ds:CreateServiceA
xor     edx, edx
lea     eax, [esp+404h+FileTime]
mov     dword ptr [esp+404h+SystemTime.wYear], edx
lea     ecx, [esp+404h+SystemTime]
mov     dword ptr [esp+404h+SystemTime.wDayOfWeek], edx
push    eax             ; lpFileTime
mov     dword ptr [esp+408h+SystemTime.wHour], edx
push    ecx             ; lpSystemTime
mov     dword ptr [esp+40Ch+SystemTime.wSecond], edx
mov     [esp+40Ch+SystemTime.wYear], 2100
call    ds:SystemTimeToFileTime
push    0               ; lpTimerName
push    0               ; bManualReset
push    0               ; lpTimerAttributes
call    ds:CreateWaitableTimerA
push    0               ; fResume
push    0               ; lpArgToCompletionRoutine
push    0               ; pfnCompletionRoutine
lea     edx, [esp+410h+FileTime]
mov     esi, eax
push    0               ; lPeriod
```

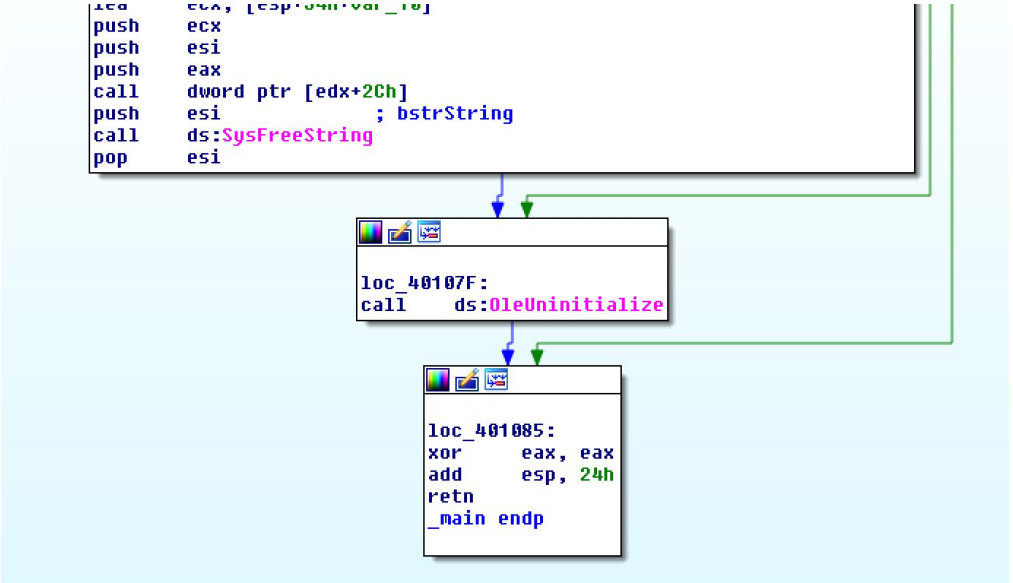
And then call CreateThread for 20 times to open website “http://www.malwareanalysisbook.com”. So we can infer it is scheduling an attack to that server.



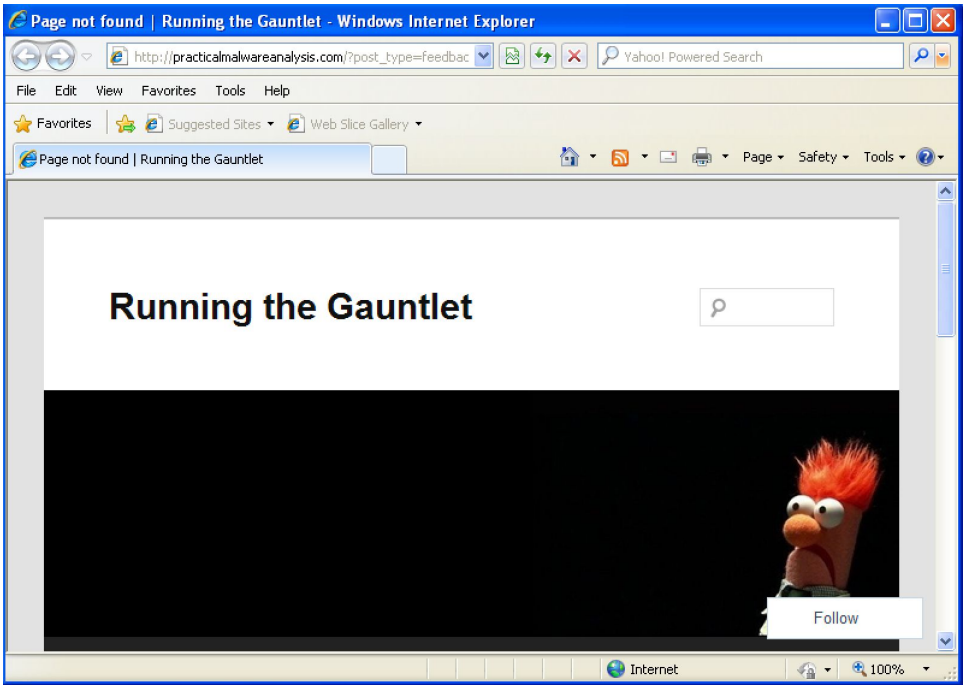
6. The threads created in this program are executed in an infinity loop, which means this program will never finish.

Lab07-02.exe:

1. This program finishes after poping up a web page. No signs shows it achieves persistence.



2. It is used to pop up a AD web page with IE. The URL is “http://www.malwareanalysisbook.com/ad.html”.




```
lea     ecx, [esp+24h+pvarg]
push    esi
push    ecx             ; pvarg
call    ds:VariantInit
push    offset psz      ; "http://www.malwareanalysisbook.com/ad.h"...
mov     [esp+2Ch+var_10], 3
mov     [esp+2Ch+var_8], 1
call    ds:SysAllocString
lea     ecx, [esp+28h+pvarg]
mov     esi, eax
```

3.The program finishes immediately after opening the web page.

Lab07-03:

1. As is shown, in this part the program is copying Lab07-03.dll to C:\\windows\\system32\\kerne132.dll.

```
loc_4017D4:
mov     ecx, [esp+54h+hObject]
mov     esi, ds:CloseHandle
push    ecx             ; hObject
call    esi ; CloseHandle
mov     edx, [esp+54h+var_4]
push    edx             ; hObject
call    esi ; CloseHandle
push    0               ; bFailIfExists
push    offset NewFileName ; "C:\\windows\\system32\\kerne132.dll"
push    offset ExistingFileName ; "Lab07-03.dll"
call    ds:CopyFileA
test    eax, eax
push    0               ; int
jnz     short loc_401806
```

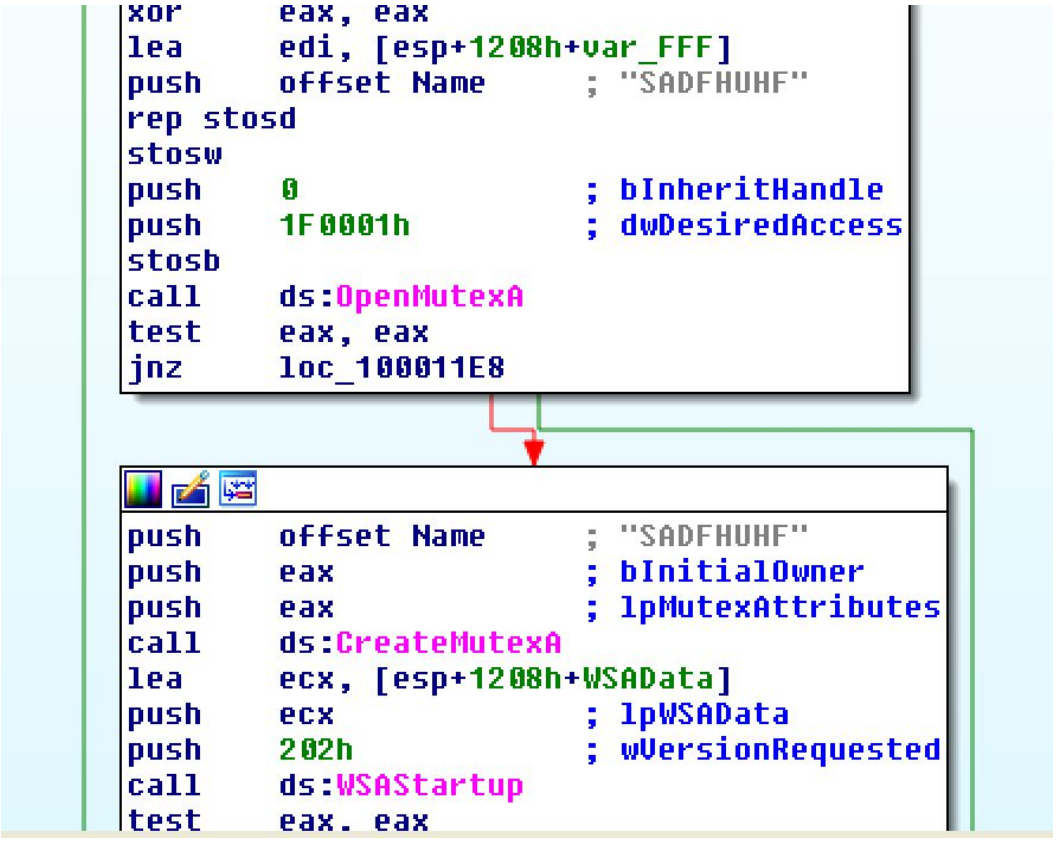
In the sub function it is searching for all files with extension “.exe”.

```
mov     edi, ebp
lea     ebx, [esp+ecx+154h+FindFileData.dwReserved1]
or      ecx, 0FFFFFFFFh
repne scasb
not     ecx
dec     ecx
lea     edi, [esp+154h+FindFileData.cFileName]
mov     edx, ecx
or      ecx, 0FFFFFFFFh
repne scasb
not     ecx
dec     ecx
lea     eax, [edx+ecx+1]
push    eax             ; Size
call    ds:malloc
mov     edx, [esp+158h+lpFileName]
mov     ebp, eax
mov     edi, edx
or      ecx, 0FFFFFFFFh
xor     eax, eax
push    offset a_exe    ; ".exe"
repne scasb
not     ecx
```

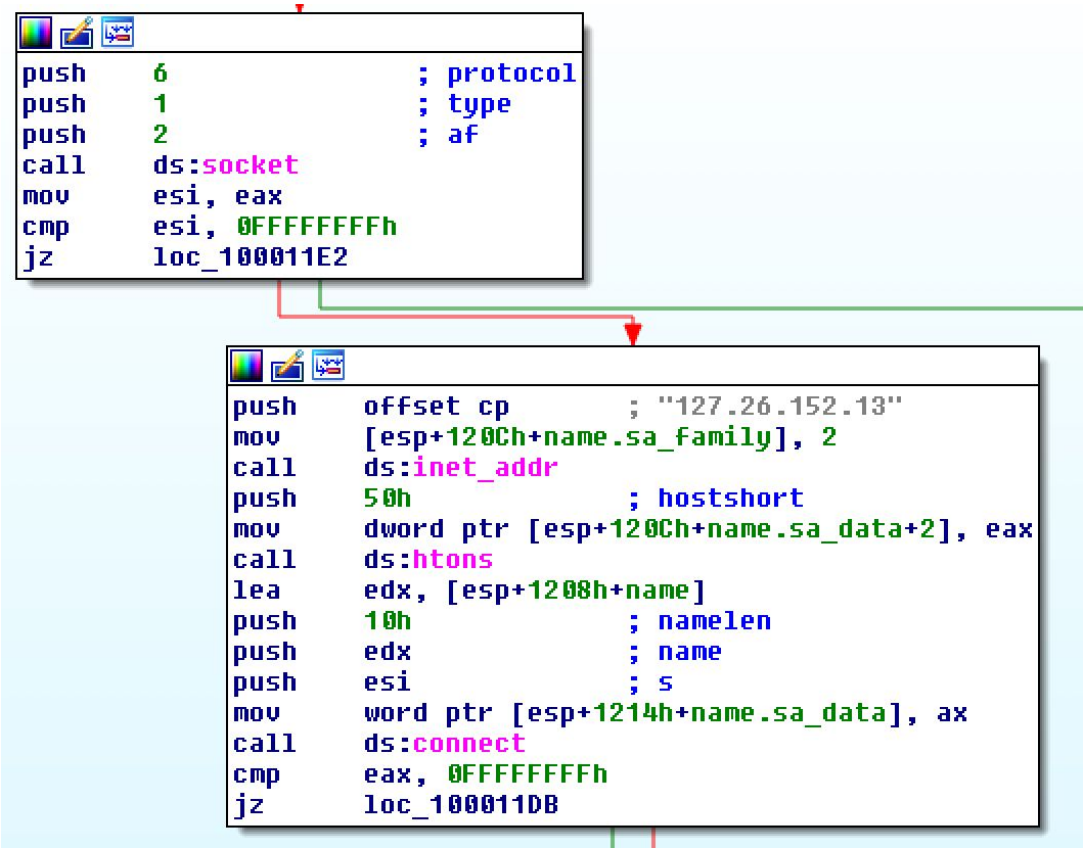
Given the fact that it is calling function CreateFileMappingA and function MapViewOfFile, which are 2 functions used to map file into memory, we can infer that this program is creating a fake Kernel32.dll, which is called kerne132.dll, and make it imported by all .exe files.

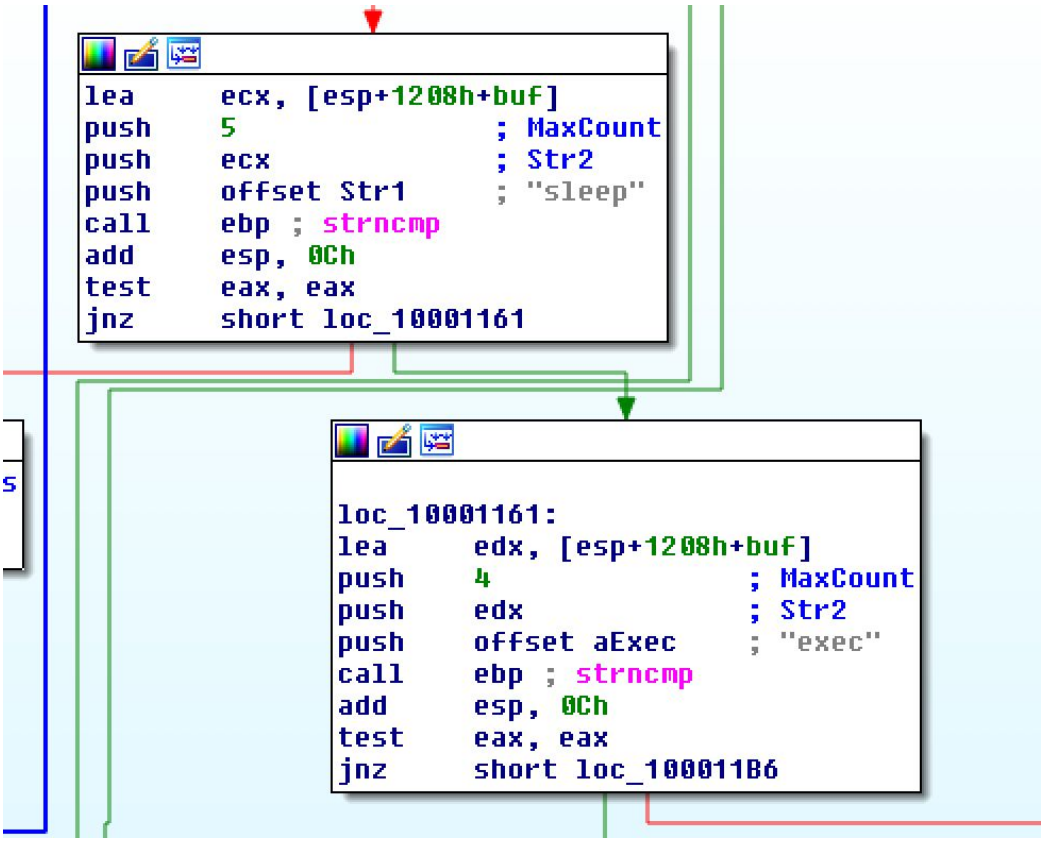
```
push    1               ; dwShareMode
push    10000000h       ; dwDesiredAccess
push    eax             ; lpFileName
call    ds:CreateFileA
push    0               ; lpName
push    0               ; dwMaximumSizeLow
push    0               ; dwMaximumSizeHigh
push    4               ; flProtect
push    0               ; lpFileMappingAttributes
push    eax             ; hFile
mov     [esp+34h+var_4], eax
call    ds:CreateFileMappingA
push    0               ; dwNumberOfBytesToMap
push    0               ; dwFileOffsetLow
push    0               ; dwFileOffsetHigh
push    0F001Fh         ; dwDesiredAccess
push    eax             ; hFileMappingObject
mov     [esp+30h+hObject], eax
call    ds:MapViewOfFile
mov     esi, eax
test    esi, esi
```

2. A good signature is that it is using filename “kerne132.dll”. Another signature is that it is creating a mutex named “SADFHUHF”.



3. This program is establishing a connection with a remote host and to follow instructions from the host.





4. It is hard to remove since it make modification to every .exe file. One way could be modify the kerne132.dll file created by this malware to make it no longer malicious and therefore will do no harm to the system.