

## M7T2. Tarea 2

### Reto Atenea: Wannacry:

Para el análisis de Wannacry he usado dos herramientas: la herramienta Pev dentro de la distribución de Linux Remnux y Ida Pro.

Usando Pev he buscado por strings que contengan los caracteres http (En la página del reto nos indican que busquemos el dominio)

```
remnux@remnux:~/Downloads$ pestr wannacry.exe | grep http
http://www.ccncertnomorecryaadrifaderesddferrrqdfwa.com
Licensed to The Apache Software Foundation http://www.apach
```

Nos devuelve la URL:

<http://www.ccncertnomorecryaadrifaderesddferrrqdfwa.com>

Sacando el md5 de la url y subiéndolo a Atenea podemos comprobar si es el flag correcto o no:



También he revisado wannacry en ida pro para ver algo más del comportamiento:

El string nos sale directamente en las strings que detecta el programa:

Address	String
0x004313d0	http://www.ccncertnomorecryaadrifaderesddferrrqdfwa.com
0x0071d41c	%s %s HTTP/1.0\r\n%s%sContent-length: %u\r\nContent-type: %s\r\n%s\r\n
0x0071d478	%s %s HTTP/1.0\r\n%s%sContent-length: %u\r\nContent-type: %s\r\n%s\r\n
0x0071e97c	HTTP

Y podemos como llama a algunas funciones como InternetOpen:

```
mov ecx, 0xe ; 14
mov esi, str.http://www.ccncertnomorecryaadrifaderesddferrrqdfwa.com ; 0x4313d0
lea edi, [var_8h]
xor eax, eax
rep movsd dword es:[edi], dword ptr [esi]
movsb byte es:[edi], byte ptr [esi]
mov dword [var_41h], eax
mov dword [var_45h], eax
mov dword [var_49h], eax
mov dword [var_4dh], eax
mov dword [var_51h], eax
mov word [var_55h], ax
push eax
push eax
push eax
push 1 ; 1
push eax
mov byte [var_6bh], al
call dword [InternetOpenA] ; 0x40a134
push 0
push 0x00000000
```

## RansomWare:

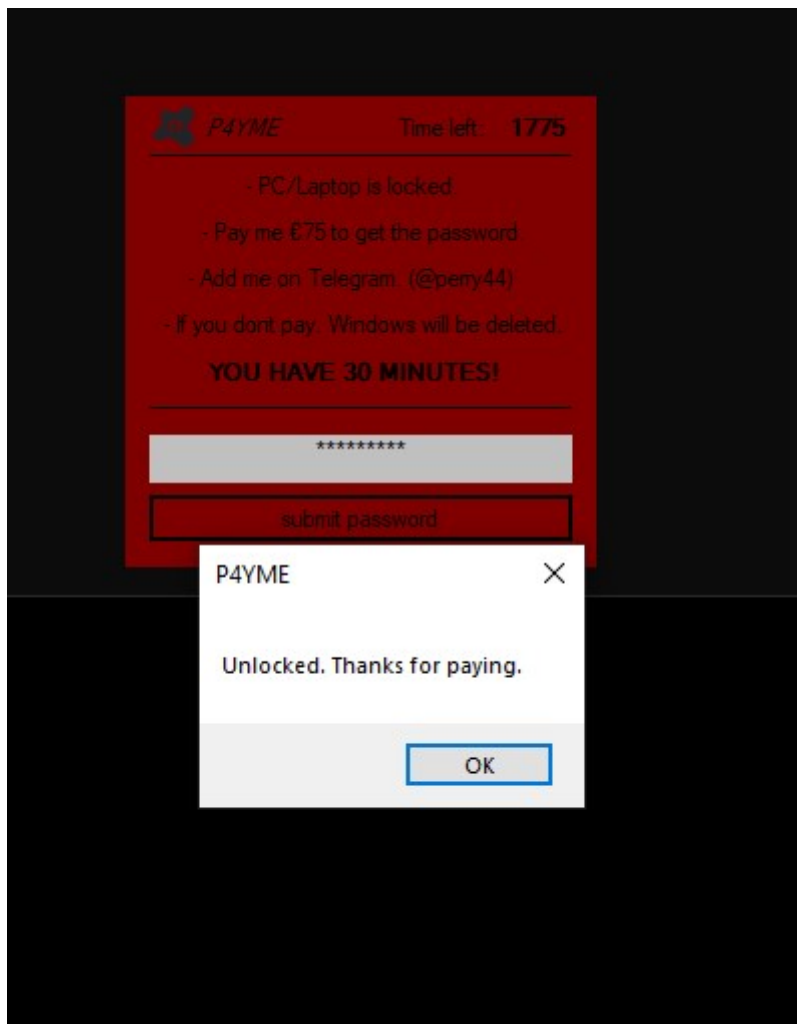
El ransomware lo he analizado con la herramienta Pev, que se encuentra en la distro de Remnux. Por lo que he visto es una herramienta muy potente a la hora de detectar strings en los programas, ya que directamente me ha detectado la contraseña para escapar del ransomware. Cosa que otros programas como Cutter o Ida Pro no han conseguido:

Este es el resultado de usar el comando: pestr sobre el programa.

```
7/7/2021  
ssadmin delete shadows  
/all  
1800  
M4St3RcB$  
ipconfig  
/release  
Unlocked. Thanks for paying.  
P4YME  
explorer.exe
```

Contraseña: M4St3RcB\$

Y el resultado en una VM al ejecutar el código malicioso y meter la contraseña:



## CracMe01.exe

He usado la misma técnica que en casos anteriores para analizar que strings tiene el programa:

Usando el comando pestr he obtenido lo siguiente:

Algunas strings interesantes:

```
j:PUp3
"U<4(|8Y9!
Hola bienvenido al primer reto.
Introduce la contrase
a para superarlo:
Reto superado, guarda la password para las respuestas
Prueba otra vez!!!!
```

Algunas funciones que usa el programa:

```
api-ms-win-err-heap-l1-1-0.dll
UnhandledExceptionFilter
SetUnhandledExceptionFilter
GetCurrentProcess
TerminateProcess
IsProcessorFeaturePresent
QueryPerformanceCounter
GetCurrentProcessId
GetCurrentThreadId
GetSystemTimeAsFileTime
InitializeSListHead
IsDebuggerPresent
GetModuleHandleW
KERNEL32.dll
<?xml version='1.0' encoding='UTF-8' standalone='yes'?>
<assembly xmlns='urn:schemas-microsoft-com:asm.v1' manifestVersion='1.0'>
  <trustInfo xmlns="urn:schemas-microsoft-com:asm.v3">
    <security>
      <requestedPrivileges>
        <requestedExecutionLevel level='asInvoker' uiAccess='false' />
      </requestedPrivileges>
```

He intentado solucionarlo a partir de aquí con toda la suite de programas que hemos ido usando en el modulo pero no he conseguido sacar la contraseña.