## Benchmark EPICODE W24D4

Ci è stato chiesto di analizzare il malware presente nella cartella presente all'interno della macchina vortuale chiamato Build\_Week\_Unit\_3 e di rispondere a vari quesiti facendo Analisi statica e analisi dinamica.

Come prima cosa ho avviato IDA PRO a prendo il Malware.

I parametri riscontrati nella funzione Main() sono 3 (int arg, const char \*\*argv, const char \*\*envp) e si identificano 8 variabili in totale (hModule, Data, var\_117, var\_8, var\_4, argc, argv, envp) come dimostrato dallo screen sottostante.

```
; int __cdecl main(int argc, const char **argv, const char **envp)
_main proc near

hModule= dword ptr -11Ch

Data= byte ptr -118h

var_117= byte ptr -117h

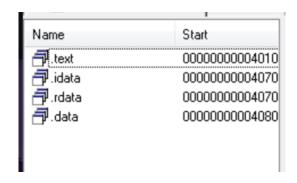
var_8= dword ptr -8

var_4= dword ptr -4

argc= dword ptr 8

argv= dword ptr 0Ch
envp= dword ptr 10h
```

Successivamente ci è stato chiesto di scoprire in quante sezione fosse suddiviso il file al suo interno. Ci sono 4 sezioni trovate facendo click su "view", open subviewers e segments (.text,.idata,.rdata,.data) dove per esempio la sezione ".rdata" contiene i dati di sola lettura mentre .idata è usata per memorizzare le funzioni da DLL esterne.



Ci è stato chiesto infine quali librerie importa il file. Le librerie che importa sono 2: kernel32.dll e advapi32.dll. Considerando che all'interno della sezione ci sono API come la LockResource, LoadResource, FindResource e SizeofResource sono portato a pensare che il Malware sia un Dropper ovvero è un programma malevolo al cui interno contiene il Malware.

000000	RegSetValueExA	ADVAPI32
<b>E</b> 000000	RegCreateKeyExA	ADVAPI32
<b>E</b> 000000	SizeofResource	KERNEL32
<b>E</b> 000000	LockResource	KERNEL32
<b>E</b> 000000	LoadResource	KERNEL32
<b>E</b> 000000	VirtualAlloc	KERNEL32
<b>E</b> 000000	GetModuleFileNameA	KERNEL32
<b>E</b> 000000	GetModuleHandleA	KERNEL32
<b>E</b> 000000	FreeResource	KERNEL32
<b>E</b> 000000	FindResourceA	KERNEL32
<b>E</b> 000000	CloseHandle	KERNEL32
<b>😭</b> 000000	GetCommandLineA	KERNEL32
<b>1 0000000</b> 00000000000000000000000000000	GetVersion	KERNEL32
🔁 000000	ExitProcess	KERNEL32
<b>🔁</b> 000000	HeapFree	KERNEL32
<b>E</b> 000000	GetLastError	KERNEL32
<b>1</b> 0000000	WriteFile	KERNEL32
<b>🔁</b> 000000	TerminateProcess	KERNEL32
<b>🔁</b> 000000	GetCurrentProcess	KERNEL32
<b>🔁</b> 000000	UnhandledExceptionFilter	KERNEL32
<b>🔁</b> 000000	FreeEnvironmentStringsA	KERNEL32
nnnnnn	FreeEnvironmentString«W/	KERNEL 32

Dopodichè sono passato all'analisi delle varie locazioni di memoria richieste.

- -Lo scopo della funzione chiamata alla locazione di memoria 00401021 è RegCreateKeyExA cioè creare una chiave di registro.
- -L'oggetto rappresentato nella locazione 00401017 è molto probabilmente la schermata di login di windows NT,XP ecc ecc.
- -Il significato delle istruzioni tra gli indirizzi 00401027 e 00401029 quindi:

test eax,eax jz short loc\_401032

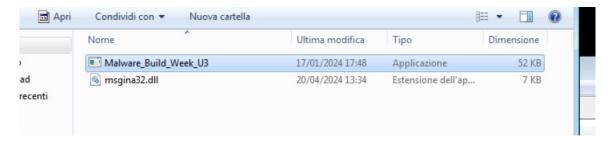
Significa che viene fatta una comparazione tra eax e eax se è 0 viene fatto un jump alla locazione 401032 con un jz ovvero jump zero.

In pseudo codice C potrebbe essere: if (eax==0) fai questo

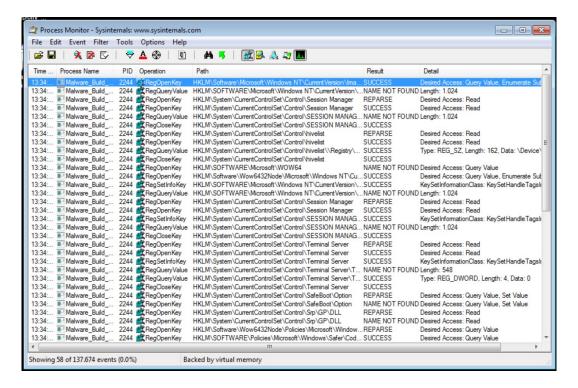
else salta a loc\_401032

il valore del parametro ValueName della locazione 00401047 è gina.dll

Infine sono passato all'analisi dinamica mediante procmon, avviandolo, resettando tutti i filtri come suggerito, avviando il malware, lasciando che agisca per un po' di tempo e stoppando la cattura. Ho riscontrato che è stato generato un file chiamato msgina32.dll, facendo ricerche in rete il nome gina.dll nello specifico viene associato alla pagina di login utente di Windows NT/XP. La mia idea attualmente è che questo file possa essere la resilienza del malware in questione, ovvero si riavvia ogni volta che viene fatto un login nella macchina, o comunque vengano caricati i dati utente.



Ci è stato chiesto di filtrare sulle attività di registro di windows

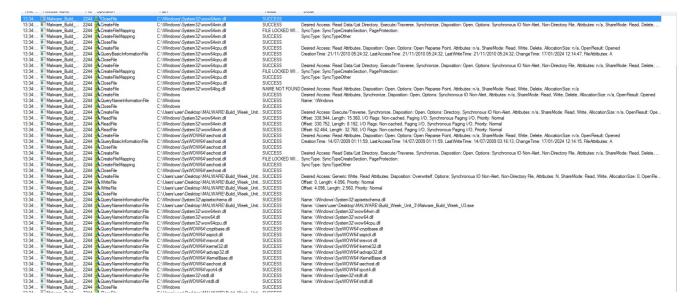


Da qui si può evincere che la chiave di registro creata e la HKLM o *HKEY\_LOCAL\_MACHINE*, che è la chiave di registro adibita a salvare le informazioni generali per tutti gli utenti presenti su una macchina ( questo rafforza i miei sospetti che il malware appunto utilizzi appunto la schermata di login utenti per ottenere resilienza sul sistema).

Aggiungendo un altro filtro su RegSetValue possiamo evincere che il valore associato a questa chiave è REG\_SZ come mostrato qua sotto.



Infine ci è stato chiesto di passare al File System e analizzare quale chiamata di sistema ha modificato il contenuto della cartella dove è presente l'eseguibile del Malware come sotto riportato.



Le chiamate principali sono CreateFile e WriteFile perché appunto creano un nuovo file all'interno della cartella.

## Considerazioni finali:

Con tutte le info a mia disposizione, sono certo che il Malware sia un Dropper ( o Trojan), vista la presenza delle API LockResource, LoadResource, FindResource e SizeofResource già nominate in precedenza, piuttosto comuni in questo genere di Malware, il fatto che crei un file chiamato msgina32.dll quindi associabile al fatto che il file gina.dll in windows è associato alla schermata di login del sistema operativo e al fatto che la chiave che utilizza nel registro e la HKLM, sono abbastanza sicuro che questo malware una volta installato ottenga la resilienza alla login dell'utente all'avvio dell'OS mediante la Gina.dll. Per supportare questa mia tesi ho consultato VirusTotal per confermare i miei sospetti e a quanto sembra sono del tutto fondati come da screen sotto.

