Progetto M6 - Lorenzo Turini

Analisi Statica

1. Nella funzione Main() sono passati 3 parametri (int argc, const char **argv, const char **envp)
2. All'interno della funzione Main () sono presenti 5 variabili. Le variabili locali sono: hModule (dword ptr -11Ch), Data (byte ptr -118h), var_117 (byte ptr -117h), var_8 (dword ptr -8), var_4
3. Il file eseguibile è composto da diverse sezioni, come .text, .data, .bss, e altre. Analizzando le sezioni presenti nel file eseguibile, è possibile identificare due sezioni importanti:
   .text: Questa sezione contiene il codice eseguibile del programma, incluse le funzioni definite nel programma.
   .data: Questa sezione contiene dati statici inizializzati, come variabili globali con valori specifici assegnati.
4. Per identificare le librerie importate dal file eseguibile, puoi analizzare la tabella delle importazioni del file. Queste librerie potrebbero includere funzioni di sistema, di rete o di gestione dei file che il malware potrebbe sfruttare per svolgere le sue attività dannose. Il Malware importa 2 librerie: 1. Kernel32.dll, 2. ADVAPI32.dll. le librerie in questione offrono al malware la possibilità di interagire con molte parti del sistema operativo, svolgendo operazioni dannose come la gestione dei processi, l'accesso ai file e la manipolazione delle impostazioni di sicurezza.

2.

1. Lo scopo della funzione 00401021 è una al RegCreateKeyExA una funzione della libreria advapi32.dll del sistema operativo Windows che viene utilizzata per creare o aprire una chiave del registro di sistema. Questa stringa è utilizzata per specificare una chiave del registro di sistema di Windows che contiene impostazioni per il sistema operativo, in particolare per la funzione RegCreateKeyExA se chiamata subito dopo questa istruzione.

2.Nella chiamata alla funzione RegCreateKeyExA alla locazione 0x00401021 nel codice assembly, i parametri vengono passati alla funzione attraverso lo stack. Questo è un metodo comune di passaggio dei parametri in linguaggio assembly e in molte altre architetture di computer. Prima della chiamata alla funzione RegCreateKeyExA, ci sono diverse istruzioni push nel codice; Dopo aver inserito tutti i parametri sullo stack, il codice esegue "call ds:RegCreateKeyExA" alla locazione 0x00401021. Questo indica all'assembly di chiamare la funzione RegCreateKeyExA con i parametri precedentemente pushati sullo stack. Infine, alla chiamata della funzione RegCreateKeyExA verranno letti i parametri dal fondo dello stack nell'ordine inverso rispetto a come sono stati inseriti (cioè, hKey, SubKey, Reserved, lpClass, dwOptions, samDesired) e li userà per eseguire l'operazione richiesta.

3. L'istruzione Assembly all'indirizzo 00401017 mostra che un valore viene spinto nello stack. In particolare, l'istruzione push sta spingendo un indirizzo di memoria (offset) relativo alla stringa SubKey nello stack. Il malware potrebbe utilizzare la chiave di registro specificata per creare o modificare impostazioni chiave nel sistema operativo. Il tentativo di manipolare le chiavi da parte del malware comporterebbe uno stato di persistenza del suddetto all'interno del Sistema Operativo.

4. Le istruzioni assembly fra le comprese tra i due indirizzi proposti dal quesito sono:

```
.text:00401027          test    eax, eax
```

Questa istruzione esegue un'operazione AND tra il registro eax e sé stesso (eax). In pratica, questa istruzione controlla se eax presenta un valore uguale a zero. L'operazione AND tra un registro e sé stesso mantiene il registro invariato modificando, a seconda del risultato, solo il flag ZF (Zero Flag).

```
.text:00401029          jz      short loc_401032
```

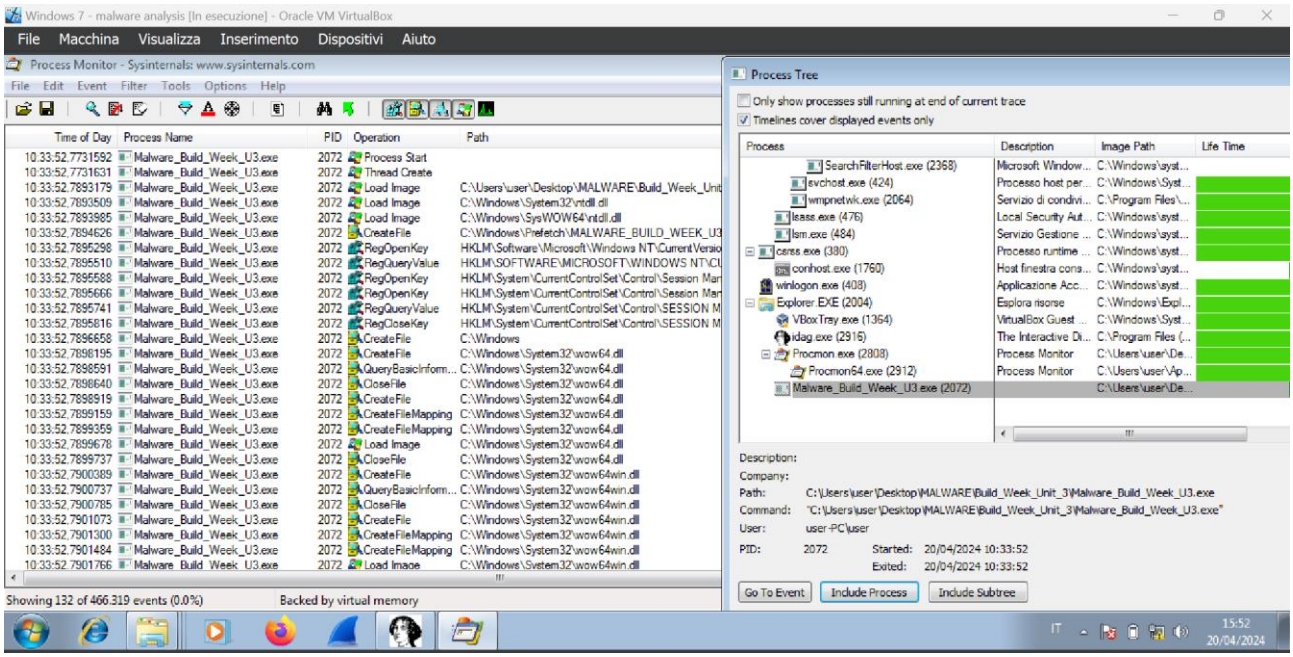
L’istruzione in questione corrisponde ad un salto condizionale. Se il flag ZF (Zero Flag) è impostato, il codice eseguirà un salto all'indirizzo specificato (loc_401032). Il flag ZF è impostato se il risultato dell'istruzione test eax, eax corrisponde a zero, il che significa che il valore nel registro **eax** era uguale a zero

5.
```
if (eax == 0) {
   goto loc_401032;
}
```

6. Il parametro ValueName passato alla funzione RegSetValueExA alla locazione 00401047 è "GinaDLL". Questo è il nome del valore da impostare nella chiave di registro indicata da hKey. La funzione RegSetValueExA verrà utilizzata per impostare un valore con il nome "GinaDLL" nella chiave di registro specificata.

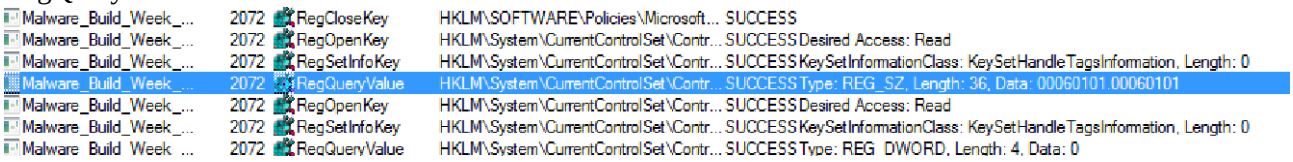
Analisi dinamica

All’interno della cartella del malware, una volta eseguito, è stata creato il file Malware_Build_Week_U3.$$$ è sospetto a causa del suo nome generico che include il termine "Malware" e l'uso di un'estensione inusuale. I file con estensioni simili possono essere utilizzati dai malware per memorizzare dati temporanei o informazioni di configurazione, o per eseguire ulteriori parti dell'attacco. Questo file potrebbe contenere codice eseguibile, dati di configurazione del malware, o altre informazioni utilizzate dal malware durante la sua esecuzione. Si nota inoltre la creazione del file msgina32.dll risulta essere un tentativo del malware di compromettere il sistema operando in modo simile a msgina.dll, libreria di autenticazione di Windows. Il malware potrebbe utilizzare msgina32.dll per garantire persistenza, alterare il comportamento del sistema, o per nascondere la sua presenza.



Dopo aver impostato il filtro, come richiesto dalla consegna da Procmon, si nota la creazione della chiave di registro RegOpenKey è una funzione delle API di Windows che consente di aprire una chiave di registro esistente nel registro di sistema di Windows. La funzione RegOpenKey viene chiamata con un handle al registro (hKey) che rappresenta una radice del registro (come HKEY_LOCAL_MACHINE o HKEY_CURRENT_USER) oltre ad una stringa che ne specifica il percorso. In questo caso il malware userebbe RegOpenKey per aprire le chiavi di registro al fine di modificare impostazioni di sistema, mantenere persistenza o eseguire altre operazioni dannose.

Il valore associato alla chiave di registro RegOpenKey corrisponde a 36 come si può vedere dall'operazione RegQueryValue.



La chiamata di sistema che ha modificato la cartella dove è situato l'eseguibile del malware risulta essere la seguente:

Riprendendo i dati dell'analisi statica e dinamica è possibile delineare un profilo piuttosto specifico del malware che potrebbe presentarsi, in base ai dati raccolti, o come un dropper che è un tipo di malware progettato per installare o "rilasciare" altri malware sul sistema bersaglio. Spesso è il primo stadio di un'infezione e può scaricare e installare altri programmi dannosi, il dropper può creare o modificare file sul sistema, potrebbe scaricare e installare ulteriori malware, potrebbe alterare le chiavi di registro per eseguire i file rilasciati all'avvio del sistema cercando di nascondere la sua presenza. Se fosse invece un keylogger creerebbe anche una persistenza installando un file sul sistema operativo "camuffandosi" con un nome simile al file di sistema per poter accedere o creare o modificare le chiavi di registro del sistema per mantenere esecuzione e presenza creando così una persistenza sul sistema.

Il malware in questione sembrerebbe corrispondere a quest'ultimo profilo, quindi un keylogger; indipendentemente dalla tipologia di malware il sistema risulta infetto e sarebbe opportuno rimuovere la libreria msgina32.dll.