



MALWARE ANALYSIS

INTRODUZIONE

Nel panorama digitale odierno, dove la tecnologia permea ogni aspetto della nostra vita, la sicurezza informatica assume un ruolo di primaria importanza. La minaccia costante rappresentata dai malware, software dannosi con scopi malevoli, incombe su individui, organizzazioni e infrastrutture critiche. La loro capacità di infiltrarsi nei sistemi, sottrarre dati sensibili, arrecare danni finanziari e compromettere la privacy degli utenti rende l'analisi del malware un'esigenza impellente, una linea di difesa in continua evoluzione.

L'IMPORTANZA DELL'ANALISI MALWARE

L'analisi del malware non è un semplice esercizio accademico, ma un'attività cruciale per la sicurezza informatica. Comprendere il funzionamento di questi software nocivi permette di:

- **Identificare e classificare le minacce:** Decodificare la natura del malware, riconoscerne le sue caratteristiche e il suo modus operandi è il primo passo per contrastarlo efficacemente.
- **Sviluppare strumenti di difesa:** Le informazioni estratte dall'analisi alimentano la creazione di software antivirus e firewall in grado di individuare e neutralizzare le nuove minacce in modo tempestivo.
- **Comprendere le tattiche degli attaccanti:** Studiando il comportamento dei malware, si possono comprendere le strategie e le tecniche impiegate dagli hacker, anticipando i loro futuri attacchi.
- **Proteggere la privacy e l'integrità dei dati:** Contrastare i malware significa salvaguardare dati sensibili, informazioni finanziarie e la privacy degli utenti, elementi essenziali per la sicurezza individuale e collettiva.

A tal proposito questo documento si propone di intraprendere un'analisi approfondita di un malware specifico, svelandone i segreti e comprendendone il modus operandi. Attraverso un'indagine meticolosa, il documento mira a:

- **Identificare la natura del malware:** Decodificare il tipo di malware, le sue caratteristiche e le sue finalità è il primo passo per contrastarlo efficacemente.
- **Comprendere il suo comportamento:** Osservare come il malware interagisce con il sistema, quali dati attacca e come si diffonde permette di anticipare le sue mosse.
- **Svelare le sue tecniche:** Analizzare il codice del malware, le sue strategie di offuscamento e i suoi meccanismi di persistenza aiuta a sviluppare strumenti di difesa più efficaci.
- **Valutare il suo impatto:** Misurare il potenziale danno causato dal malware permette di definire le priorità di intervento e le misure di contenimento.

ANALISI SICURA IN AMBIENTE ISOLATO

La sicurezza è un fattore imprescindibile nell'analisi del malware. Affrontare un nemico così insidioso richiede precauzioni adeguate per evitare di mettere a rischio il proprio sistema e la propria rete. Per questo motivo, l'intera analisi del malware verrà condotta all'interno di un ambiente protetto e isolato, una macchina virtuale Windows 7 appositamente allestita e sconnessa dalla macchina host.

IMPORTANZA DI UN AMBIENTE ISOLATO

L'utilizzo di una macchina virtuale offre molteplici vantaggi in termini di sicurezza:

- **Confinamento del Malware:** La macchina virtuale crea un ambiente separato e autonomo, impedendo al malware di interagire con il sistema host o con la rete reale. In caso di infezione, il malware rimane confinato all'interno della macchina virtuale, evitando danni al sistema principale.
- **Facilità di Ripristino:** Se l'analisi provoca danni alla macchina virtuale, è possibile ripristinarla facilmente a uno stato precedente senza conseguenze per il sistema host. Questo permette di sperimentare e analizzare il malware senza timore di compromettere il proprio sistema operativo o i dati personali.
- **Protezione della Rete:** L'isolamento della macchina virtuale dalla rete reale previene la diffusione del malware ad altri dispositivi o sistemi connessi. In questo modo, si evita il rischio di epidemie informatiche e si protegge la sicurezza della rete.

ANALISI MALWARE

Durante lo studio dell'analisi dei malware, vengono utilizzate due principali tecniche:

- **Analisi Statica**
- **Analisi dinamica**

Queste metodologie sono fondamentali per comprendere e contrastare le minacce informatiche.

ANALISI STATICA DEI MALWARE

L'analisi statica si concentra sull'esame di un eseguibile senza eseguirlo effettivamente. Questo approccio fornisce informazioni preliminari sul potenziale comportamento dannoso di un file eseguibile. Le principali caratteristiche includono:

- **Analisi Statica Basica:** Consiste nell'osservare le caratteristiche esterne del file, come le firme digitali e le stringhe incorporate, per identificare rapidamente le caratteristiche sospette o malevole.
- **Analisi Statica Avanzata:** Richiede competenze di reverse engineering per esaminare le istruzioni binarie del file utilizzando strumenti come disassemblatori. Questo approccio rivela dettagli sulle funzioni interne del malware.

ANALISI DINAMICA DEI MALWARE

L'analisi dinamica coinvolge l'esecuzione del malware in un ambiente controllato per osservarne il comportamento reale. Le caratteristiche principali includono:

- **Analisi Dinamica Basica:** Il malware viene eseguito in un ambiente isolato e monitorato per osservarne le azioni immediate, come la comunicazione di rete e le modifiche al sistema.
- **Analisi Dinamica Avanzata:** Richiede l'uso di debugger e strumenti avanzati per esaminare il comportamento interno del malware durante l'esecuzione, identificando le tecniche di evasione e le attività di sistema.

TECNICHE COMPLEMENTARI

L'analisi statica e dinamica sono complementari e vengono utilizzate insieme per ottenere una comprensione completa dei malware. L'analisi statica fornisce informazioni iniziali sui comportamenti sospetti, mentre l'analisi dinamica conferma e approfondisce queste informazioni mediante l'esecuzione controllata del malware. Questa combinazione consente di identificare e mitigare efficacemente le minacce informatiche.

ANALISI STATICA BASICA

L'analisi statica rappresenta il primo passo fondamentale per svelare i segreti di un malware. Attraverso un'esplorazione meticolosa del codice e delle risorse del malware, senza eseguirlo, è possibile ottenere informazioni preziose sulla sua natura, sulle sue funzionalità e sul suo potenziale impatto. In questa fase preliminare, ci avvarremo di quattro strumenti essenziali:

- **Md5deep:** Un potente alleato per l'identificazione univoca. Md5deep genera un'impronta digitale univoca, un hash MD5, che permette di distinguere il malware da altri file e di tracciarne la diffusione.
- **Strings Utility:** Un rivelatore di segreti nascosti. Strings Utility estrae le stringhe di testo incorporate nel malware, svelando potenziali messaggi di errore, nomi di dominio dannosi o informazioni utili per la sua identificazione.
- **CFF Explorer:** Un'immersione nelle profondità del PE. CFF Explorer ci permette di esplorare la struttura del file eseguibile PE (Portable Executable) del malware, analizzando sezioni, header e altre informazioni cruciali per comprenderne il funzionamento.
- **ExeInfoPE:** Un cacciatore di cave esperto. ExeInfoPE si distingue per la sua capacità di individuare le cave all'interno del codice del malware. Le cave, ovvero porzioni di codice vuote o commentate, possono essere utilizzate dagli sviluppatori di malware per nascondere funzionalità dannose o per confondere gli analisti. ExeInfoPE ci aiuta a smascherare queste insidie e a ottenere una visione più completa del malware.

Combinando le informazioni estratte da questi strumenti, costruiremo un solido quadro del malware, preparando il terreno per un'analisi più approfondita e per lo sviluppo di strategie di contrasto efficaci.

MD5DEEP

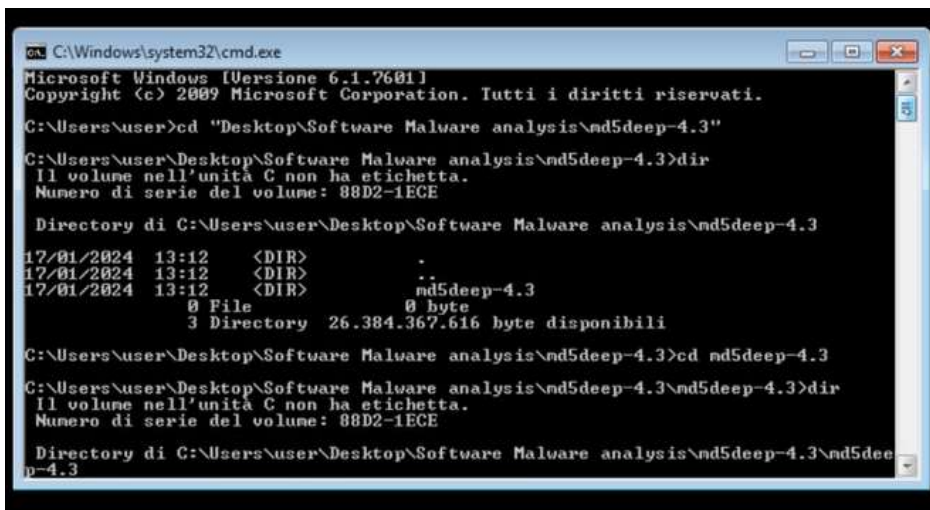
Prima di immergerci nell'analisi approfondita del malware, è fondamentale accertarne la natura dannosa. La firma digitale del file, ovvero l'hash MD5, rappresenta un'impronta digitale univoca che ci permette di identificare il malware e di verificarne la presenza nei database degli antivirus. **Md5deep** è un software ampiamente utilizzato nel mondo della sicurezza informatica, dell'amministrazione di sistema e della computer forensics. La sua funzione principale è quella di generare hash MD5, impronte digitali univoche che permettono di identificare file, software e, nel nostro caso, malware. **Md5deep** calcola un hash MD5 per ogni file indicato. L'hash MD5 è una stringa alfanumerica di 32 caratteri che rappresenta una sorta di "impronta digitale" del file. La caratteristica fondamentale di questa impronta è l'unicità: due file differenti, anche se apparentemente identici, avranno sempre hash MD5 distinti.

L'utilizzo di **Md5deep** nell'analisi del malware offre diversi vantaggi:

- **Identificazione univoca:** L'hash MD5 permette di distinguere un malware da altri file e di tracciarne la diffusione.
- **Verifica di integrità:** Confrontando l'hash MD5 di un file con quello originale, è possibile verificarne l'integrità e accertarsi che non sia stato modificato o corrotto.
- **Ricerca di informazioni:** L'hash MD5 può essere utilizzato per cercare informazioni sul malware in database online, come VirusTotal, che contengono informazioni sulla sua natura, sulle sue minacce e sui metodi di rimozione.

GENERARE L'HASH MD5 CON MD5DEEP

1. **Posizionarsi nella Cartella del Tool:** Il primo screenshot mostra l'utilizzo del prompt dei comandi per navigare nella cartella contenente il tool md5deep.



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Versione 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Tutti i diritti riservati.

C:\Users\user>cd "Desktop\Software Malware analysis\md5deep-4.3"

C:\Users\user\Desktop\Software Malware analysis\md5deep-4.3>dir
Il volume nell'unità C non ha etichetta.
Numero di serie del volume: 88D2-1ECE

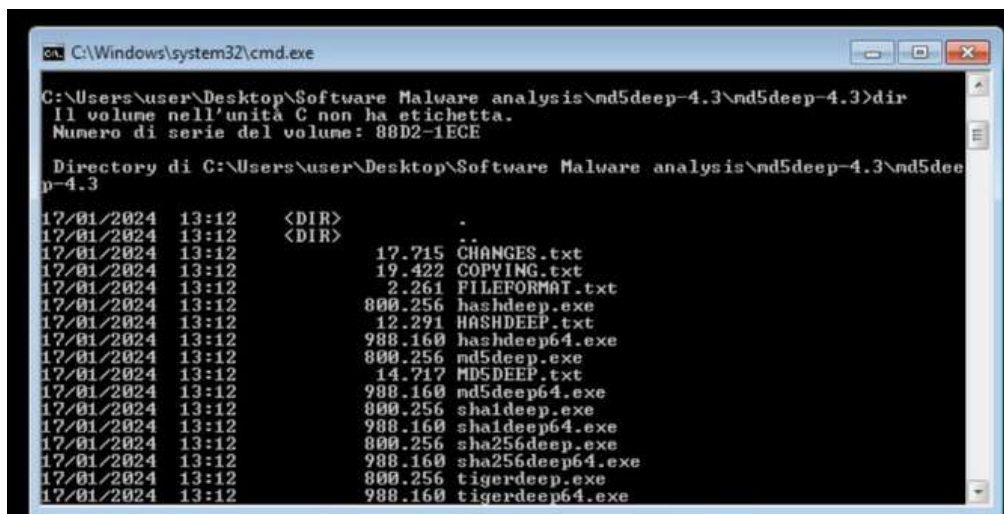
Directory di C:\Users\user\Desktop\Software Malware analysis\md5deep-4.3
17/01/2024 13:12 <DIR>      .
17/01/2024 13:12 <DIR>      ..
17/01/2024 13:12 <DIR>      md5deep-4.3
0 File             0 byte
3 Directory        26.384.367.616 byte disponibili

C:\Users\user\Desktop\Software Malware analysis\md5deep-4.3>cd md5deep-4.3

C:\Users\user\Desktop\Software Malware analysis\md5deep-4.3\md5deep-4.3>dir
Il volume nell'unità C non ha etichetta.
Numero di serie del volume: 88D2-1ECE

Directory di C:\Users\user\Desktop\Software Malware analysis\md5deep-4.3\md5dee
p-4.3
```

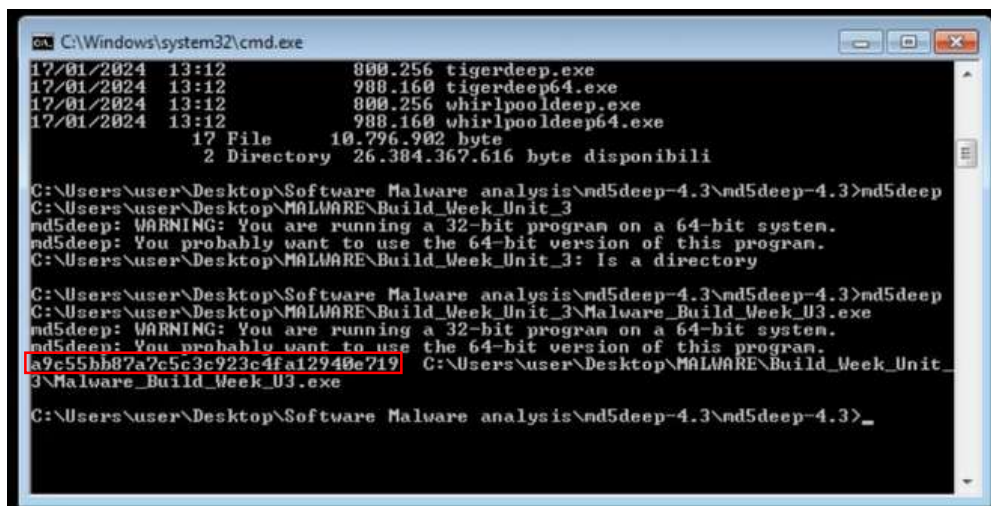

2. **Verifica del File:** Un comando "dir" elenca i file presenti nella cartella, confermando la presenza dell'eseguibile md5deep.



```
C:\Windows\system32\cmd.exe
C:\Users\user\Desktop\Software Malware analysis\md5deep-4.3>dir
Il volume nell'unità C non ha etichetta.
Numero di serie del volume: 88D2-1ECE

Directory di C:\Users\user\Desktop\Software Malware analysis\md5deep-4.3
17/01/2024  13:12    <DIR>          .
17/01/2024  13:12    <DIR>          ..
17/01/2024  13:12             17.715  CHANGES.txt
17/01/2024  13:12             19.422  COPYING.txt
17/01/2024  13:12             2.261  FILEFORMAT.txt
17/01/2024  13:12           800.256  hashdeep.exe
17/01/2024  13:12           12.291  HASHDEEP.txt
17/01/2024  13:12           988.160  hashdeep64.exe
17/01/2024  13:12           800.256  md5deep.exe
17/01/2024  13:12           14.717  MD5DEEP.txt
17/01/2024  13:12           988.160  md5deep64.exe
17/01/2024  13:12           800.256  sha1deep.exe
17/01/2024  13:12           988.160  sha1deep64.exe
17/01/2024  13:12           800.256  sha256deep.exe
17/01/2024  13:12           988.160  sha256deep64.exe
17/01/2024  13:12           800.256  tigerdeep.exe
17/01/2024  13:12           988.160  tigerdeep64.exe
```

3. **Esecuzione del Comando:** Il terzo screenshot cattura l'esecuzione del comando md5deep sul file del malware, specificando il relativo percorso, mostrando poi l'hash MD5 generato dal comando md5deep



```
C:\Windows\system32\cmd.exe
17/01/2024  13:12           800.256  tigerdeep.exe
17/01/2024  13:12           988.160  tigerdeep64.exe
17/01/2024  13:12           800.256  whirlpooldeep.exe
17/01/2024  13:12           988.160  whirlpooldeep64.exe
      17 File      10.796.902 byte
      2 Directory  26.384.367.616 byte disponibili

C:\Users\user\Desktop\Software Malware analysis\md5deep-4.3>md5deep
C:\Users\user\Desktop\MALWARE\Build_Week_Unit_3
md5deep: WARNING: You are running a 32-bit program on a 64-bit system.
md5deep: You probably want to use the 64-bit version of this program.
C:\Users\user\Desktop\MALWARE\Build_Week_Unit_3>md5deep
C:\Users\user\Desktop\MALWARE\Build_Week_Unit_3>md5deep
C:\Users\user\Desktop\MALWARE\Build_Week_Unit_3>md5deep
md5deep: WARNING: You are running a 32-bit program on a 64-bit system.
md5deep: You probably want to use the 64-bit version of this program.
a9c55bb87a7c5c3c923c4fa12940e719 C:\Users\user\Desktop\MALWARE\Build_Week_Unit_3\Malware_Build_Week_U3.exe
C:\Users\user\Desktop\Software Malware analysis\md5deep-4.3>
```

VERIFICA SU VIRUSTOTAL

Dopo aver estratto l'hash MD5 del file sospetto utilizzando il tool md5deep, siamo pronti a procedere con il passo successivo: la verifica su VirusTotal. VirusTotal è un potente alleato nella lotta contro i malware, in grado di fornire informazioni cruciali per identificare e classificare le minacce informatiche.

52 / 71

52/71 security vendors and no sandboxes flagged this file as malicious

57d8d248a8741176348b5d12dcf29f34c8f48ede0ca13c30d12e5ba0384056d7

Lab11-01.exe

Size: 52.00 KB

Last Modification Date: 15 hours ago

peexe, spreader, armadillo, checks-user-input

Reanalyze Similar More

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat label: trojan.doina/totbrick

Threat categories: trojan

Family labels: doina, totbrick, genericncq

Security vendors' analysis

AhnLab-V3	Trojan.Win32.Agent.C39204	Alibaba	Trojan:Win32/Totbrick.dfb39e83f
AliCloud	Backdoor	ALYac	Gen:Variant.Doina.65814
Antiy-AVL	Trojan.Win32.Agent	Arcabit	Trojan.Doina.D10116
Avast	Win32:Trojan-gen	AVG	Win32:Trojan-gen
Avira (no cloud)	TR/Agent.53248.465	BitDefender	Gen:Variant.Doina.65814
BitDefenderTheta	Gen:NN.Zedlaf.36802.aq4@a0clrOb	Bkav Pro	W32.AIDetectMalware
ClamAV	Win.Trojan.Agent-595082	CrowdStrike Falcon	Win/malicious_confidence_100% (W)
Cylance	Unsafe	Cynet	Malicious (score: 99)
DeepInstinct	MALICIOUS	DrWeb	BackDoor.Siggen2.1689
Elastic	Malicious (moderate Confidence)	Emsisoft	Gen:Variant.Doina.65814 (B)

L'analisi su VirusTotal ha rivelato la vera natura del file, identificandolo come un dropper/trojan. I dropper/trojan rappresentano una categoria di malware particolarmente insidiosa, poiché il loro obiettivo principale è quello di installare altri malware sul sistema infetto, fungendo da cavallo di Troia digitale.

COMPORTAMENTO TIPICO DEL DROPPER

I dropper/trojan operano in due fasi distinte:

- Infiltrazione:** Il dropper/trojan si infila nel sistema sfruttando diverse vulnerabilità, come email di phishing, siti web compromessi o allegati infetti.
- Rilascio del Payload:** Una volta all'interno del sistema, il dropper/trojan rilascia il suo vero "carico", ovvero il malware che intende installare.

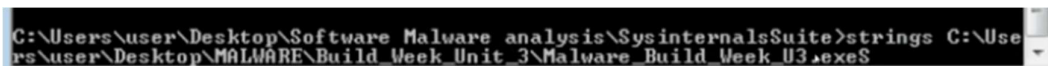
STRINGS

Gli eseguibili, spesso custodiscono al loro interno delle preziose informazioni sotto forma di stringhe. Queste stringhe possono contenere messaggi di benvenuto, dettagli sull'utilizzo del software o addirittura URL di connessioni online. Nel caso di malware, estrarre e analizzare queste stringhe può rivelarsi un'arma potentissima per smascherare i loro piani diabolici. A tal proposito, l'utility da riga di comando «strings» permette di estrarre tutte le stringhe contenute all'interno di un file eseguibile. Con un semplice comando, "strings" svela i segreti celati nel codice, permettendoci di:

- **Identificare Funzioni Sospette:** Le stringhe possono contenere nomi di funzioni API di Windows, come "regclosekey", "regsetvalueex" o "regcreatekeyw". Se queste funzioni compaiono in un contesto anomalo, potrebbero indicare un comportamento dannoso.
- **Scoprire URL Malevoli:** Le stringhe possono contenere URL a cui il malware tenta di connettersi. Questi URL potrebbero puntare a siti web che diffondono malware o a server che raccolgono dati sensibili.
- **Decifrare Messaggi Nascosti:** Le stringhe possono contenere messaggi di errore o di debug che il malware genera durante l'esecuzione. Questi messaggi possono fornire indizi preziosi sul suo funzionamento e sui suoi obiettivi.

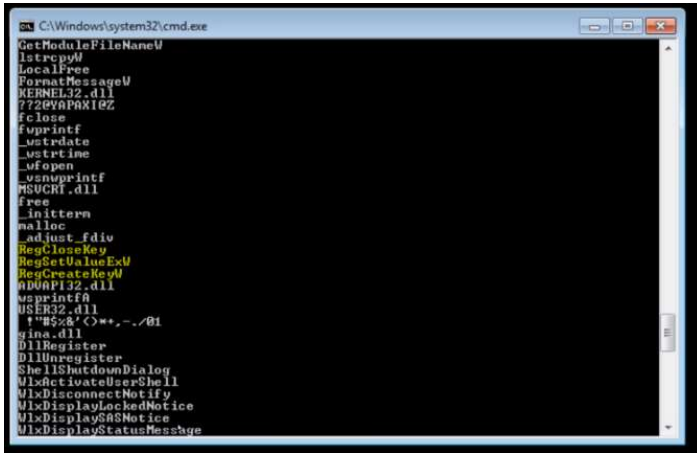
Per fornire una panoramica concisa e mirata dell'analisi, verranno presentati solo 4 screenshot rappresentativi delle stringhe estratte, escludendo l'esposizione dell'intero output generato dall'utility "strings". Questa scelta mira a focalizzare l'attenzione sugli elementi più salienti e pertinenti, facilitando la comprensione e l'interpretazione dei risultati chiave.

ESECUZIONE DEL COMANDO



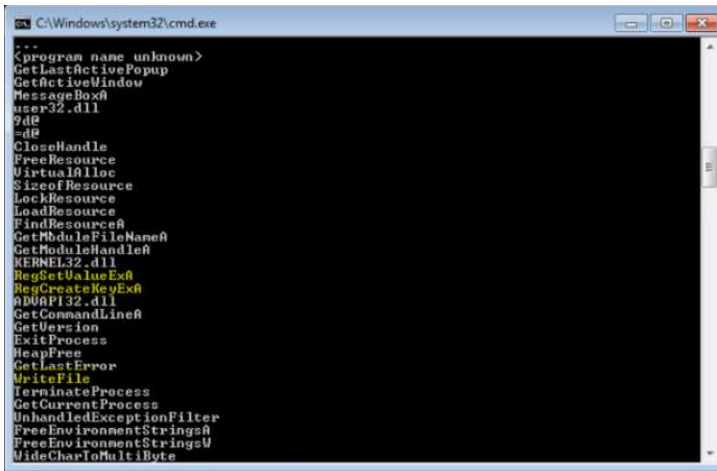
```
C:\Users\user\Desktop\Software Malware analysis\SysinternalsSuite>strings C:\Users\user\Desktop\MALWARE\Build Week Unit 3\Malware_Build Week U3.exe$
```


API REGCLOSEKEY - REGSETVALUEEX - REGCREATEKEYW



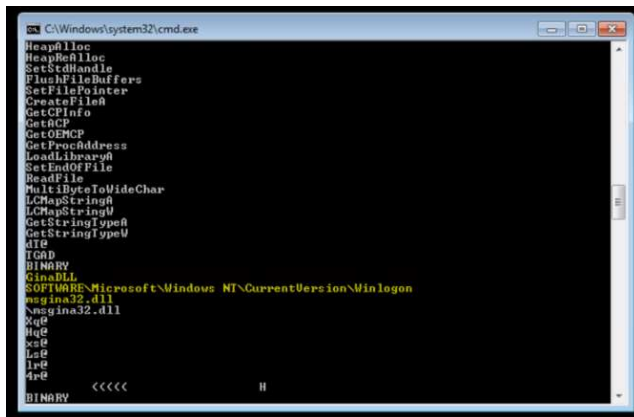
Queste stringhe suggeriscono l'utilizzo di funzioni API di Windows relative alla modifica del registro di sistema. Un uso anomalo di queste funzioni potrebbe indicare tentativi di manipolare il sistema o di installare malware.

API REGSETVALUEEX - REGCREATEKEYEX - WRITEFILE



Anche in questo caso, le stringhe indicano l'utilizzo di funzioni API di Windows per la modifica del registro di sistema e la scrittura di file. Un comportamento anomalo potrebbe essere un segnale di malware.

GINADLL - SOFTWARE\MICROSOFT\WINDOWS NT\CURRENTVERSION\WINLOGON - MSGINA32.DLL

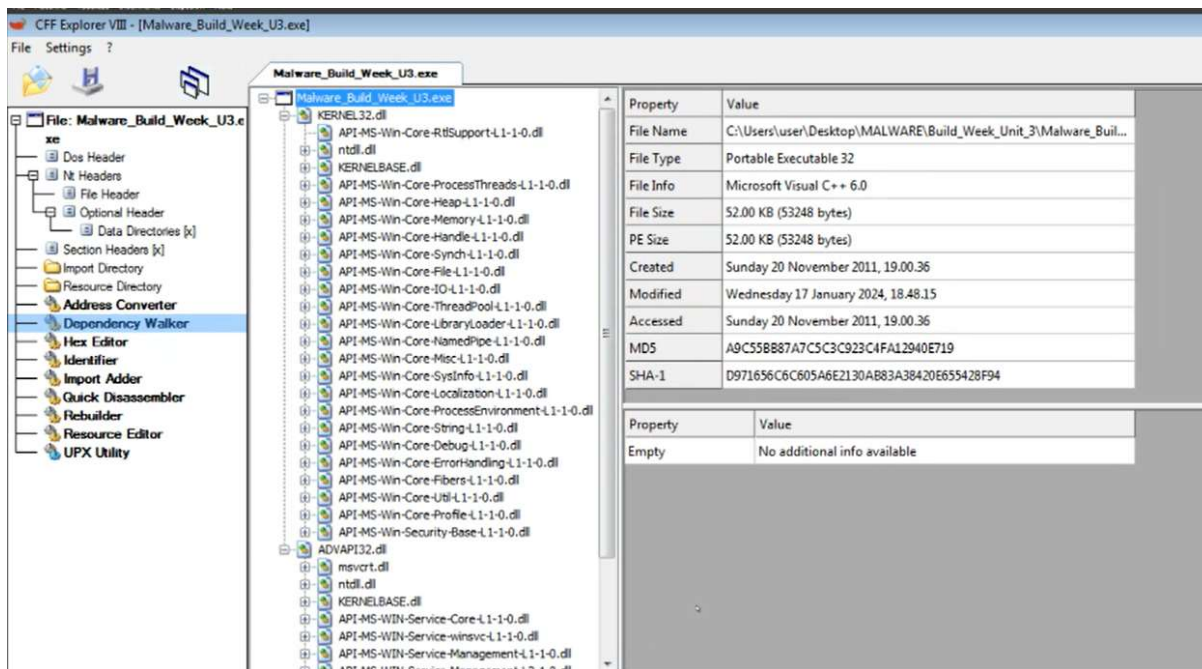


Queste stringhe potrebbero indicare il tentativo di caricare la libreria "ginadll.dll" e di interagire con il processo "winlogon". Se la libreria non è legittima, potrebbe trattarsi di un tentativo di DLL hijacking, una tecnica utilizzata dai malware per prendere il controllo del sistema.

CFF EXPLORER

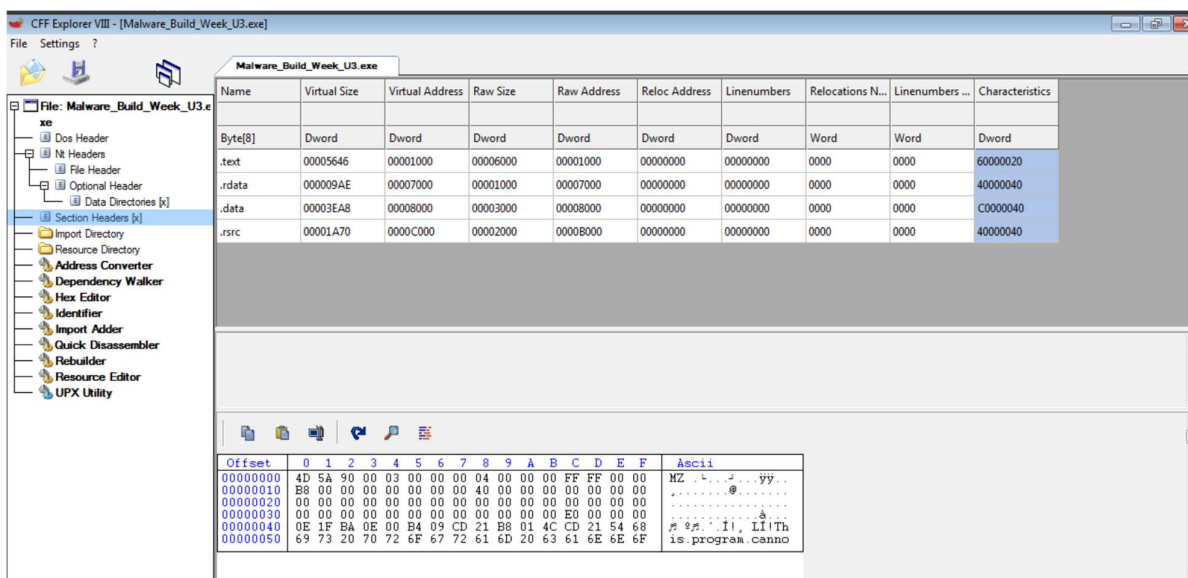
Per svelare i segreti del malware, è fondamentale esaminare le sue dipendenze esterne, ovvero le librerie e le funzioni di sistema su cui fa affidamento per funzionare. Queste informazioni preziose sono custodite all'interno dell'header del formato PE (Portable Executable) dell'eseguibile del malware. L'identificazione delle **librerie importate** dal malware è un passo cruciale per comprenderne il suo scopo e le sue potenziali minacce. Attraverso l'analisi di queste librerie, è possibile dedurre le funzionalità a cui il malware ha accesso e il suo potenziale impatto sul sistema. Oltre alle librerie, l'analisi delle **funzioni importate** dal malware fornisce ulteriori indizi sul suo comportamento. Le funzioni importate rivelano le specifiche azioni che il malware può eseguire, come la manipolazione del registro di sistema, l'accesso a file e reti o l'interazione con altri processi.

1. LIBRERIE IMPORTATE E FUNZIONI CHIAVE:



2. SEZIONI HEADER

Oltre alle librerie e alle funzioni importate ed esportate, l'analisi del formato PE (Portable Executable) del nostro malware rivela informazioni preziose attraverso le sue sezioni header. Queste sezioni strutturano l'eseguibile e ne definiscono il contenuto in modo organizzato.



Come si evince dall'immagine le quattro sezioni header utilizzate dal nostro malware sono:

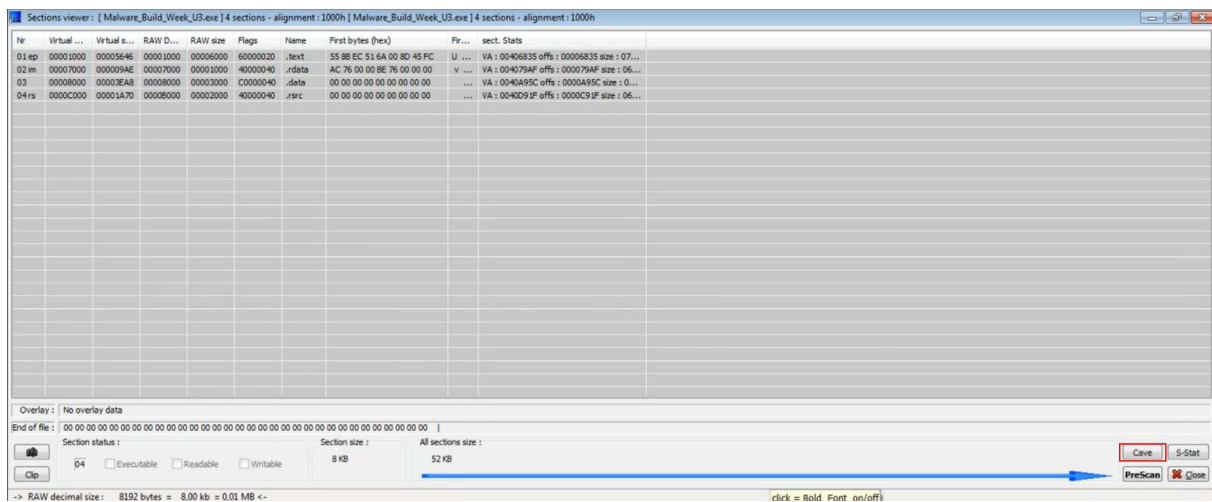
1. **.text:** La sezione .text contiene il codice eseguibile del nostro malware, ovvero le istruzioni che la CPU deve seguire per far funzionare il programma. Questa sezione è fondamentale per comprendere il comportamento del nostro malware e le sue azioni. Contiene il codice macchina, composto da istruzioni binarie che la CPU interpreta e esegue.
2. **.rdata:** La sezione .rdata contiene dati di sola lettura che il nostro malware utilizza durante l'esecuzione. Questi dati possono includere stringhe costanti, variabili globali inizializzate, dati di configurazione e altri dati statici.
3. **.data:** La sezione .data contiene dati di lettura e scrittura che il nostro malware utilizza durante l'esecuzione. Questi dati possono includere variabili locali, Variabili locali non inizializzate, dati allocati dinamicamente, buffer temporanei e altri dati dinamici.
4. **.rsrc:** La sezione .rsrc contiene le risorse del nostro malware, come icone, immagini, dialoghi e menu. Queste risorse sono utilizzate per fornire un'interfaccia utente al nostro malware e per facilitare la sua interazione con l'utente.

EXEINFOPE

ExeinfoPE condivide con CFF Explorer l'obiettivo di analizzare il formato PE (Portable Executable) dei file eseguibili, fornendo informazioni dettagliate sulle loro componenti interne. Entrambi gli strumenti permettono di esaminare le librerie importate, le sezioni header e altri elementi cruciali per comprendere il comportamento del malware.

ExeinfoPE si distingue per la sua particolare efficacia nella ricerca delle "cave" nel codice del malware. Le cave sono porzioni di codice appositamente create per nascondere dati o istruzioni dannose, rendendo difficile l'analisi statica del malware. ExeinfoPE utilizza tecniche avanzate per individuare queste cave, fornendo agli analisti un vantaggio significativo nella loro ricerca.

Lo screenshot allegato mostra la sezione delle cave del malware analizzato con ExeinfoPE. In questa sezione, ExeinfoPE elenca le cave identificate nel codice, fornendo informazioni essenziali come la loro posizione, la dimensione e il contenuto.



ANALISI DINAMICA BASICA

L'analisi dinamica basica è una tecnica di sicurezza informatica che coinvolge l'esecuzione di un software o un malware in un ambiente controllato per osservarne il comportamento e le azioni. Questo tipo di analisi fornisce informazioni immediate sulle attività del programma, come le comunicazioni di rete, le modifiche al sistema o le azioni dannose. È utile per comprendere le azioni immediate del software e identificare comportamenti sospetti. In questo scenario, ci concentreremo su quattro tool fondamentali:

- ProcMon
- Process Explorer
- RegShot
- ApateDNS

PROCMON (PROCESS MONITOR)

ProcMon offre una visione dettagliata delle attività del sistema a livello di processo, thread, registro e filesystem. Durante l'analisi del malware, ProcMon permette di:

- **Monitorare l'avvio del malware:** Osservare l'avvio del processo del malware e identificare eventuali processi secondari o dll caricate.
- **Tracciare le interazioni:** Monitorare le interazioni del malware con altri processi, come la creazione, la terminazione o l'apertura di file.
- **Analizzare le attività di registro:** Registrare le modifiche apportate al registro di sistema dal malware, fornendo indizi sulle sue tecniche di persistenza e sulle sue interazioni con il sistema.
- **Valutare l'accesso al filesystem:** Monitorare l'accesso del malware ai file e alle directory, identificando i file creati, modificati o eliminati.

PROCESS EXPLORER

Process Explorer offre una visione completa dei processi in esecuzione sul sistema, fornendo informazioni dettagliate sulle loro attività, connessioni di rete e utilizzo delle risorse. Durante l'analisi del malware, Process Explorer permette di:

- **Monitorare l'avvio del malware:** Osservare l'avvio del processo del malware e identificare eventuali processi secondari o dll caricate.
- **Tracciare le interazioni:** Monitorare le interazioni del malware con altri processi, come la creazione, la terminazione o l'apertura di file.

- **Analizzare le connessioni di rete:** Identificare le connessioni di rete aperte dal malware, i domini con cui ha comunicato e il tipo di traffico generato.
- **Valutare l'utilizzo delle risorse:** Monitorare l'utilizzo della CPU, della memoria e del disco rigido da parte del malware per identificare eventuali comportamenti anomali o consumo eccessivo di risorse.

REGSHOT

RegShot permette di creare istantanee dello stato del registro di sistema, consentendo di confrontare le modifiche apportate dal malware prima e dopo la sua esecuzione. Questa analisi rivela:

- **Modifiche alle chiavi di registro:** Identificare le chiavi di registro create, modificate o eliminate dal malware, fornendo indizi sulle sue tecniche di persistenza e sulle sue interazioni con il sistema.
- **Valutazione dell'impatto:** Valutare l'impatto potenziale delle modifiche al registro di sistema sulla stabilità e la sicurezza del sistema.

APATEDNS

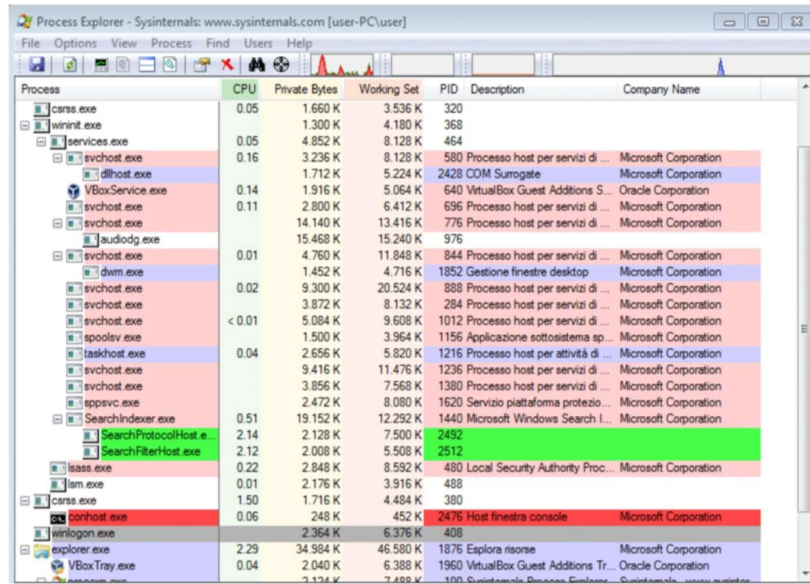
ApateDNS funge da server DNS locale, intercettando le richieste DNS del malware e reindirizzandole verso server controllati dall'analista. Questo strumento permette di:

- **Monitorare i domini a cui il malware tenta di accedere:** Identificare i domini con cui il malware ha tentato di comunicare, potenzialmente rivelando i suoi obiettivi o server di comando e controllo.
- **Bloccare connessioni dannose:** Reindirizzare le richieste DNS verso server controllati, permettendo di bloccare connessioni a domini potenzialmente dannosi o server di malware.

Attraverso una combinazione strategica di questi strumenti, è possibile ottenere una visione completa delle attività del malware e dei suoi tentativi di compromissione del sistema.

FASI DELL'ANALISI

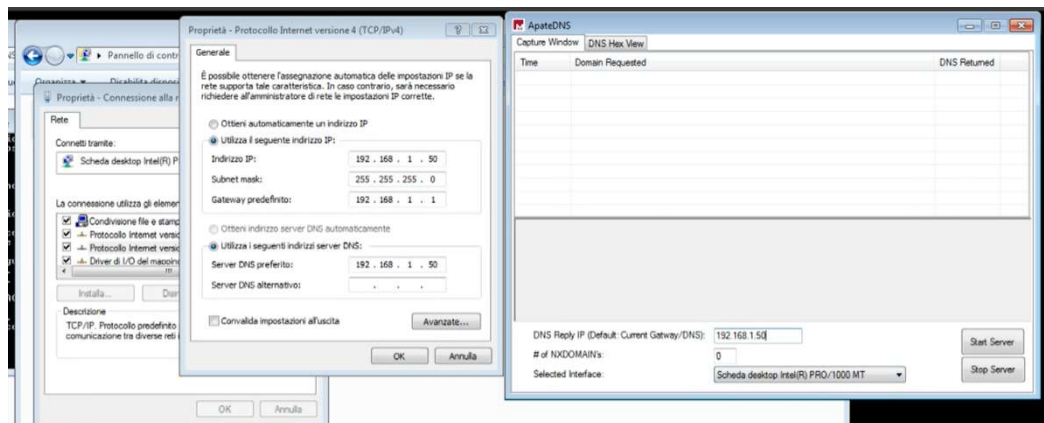
1. Avvio di Process Explorer



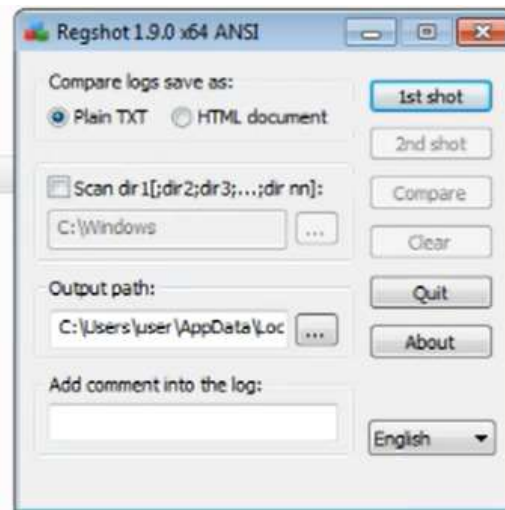
Process Explorer - Sysinternals: www.sysinternals.com [user-PC\user]

Process	CPU	Private Bytes	Working Set	PID	Description	Company Name
csrss.exe	0.05	1.660 K	3.536 K	320		
wininit.exe		1.300 K	4.180 K	368		
services.exe	0.05	4.852 K	8.128 K	464		
svchost.exe	0.16	3.236 K	8.128 K	580	Processo host per servizi di ...	Microsoft Corporation
dlhhost.exe		1.712 K	5.224 K	2428	COM Surrogate	Microsoft Corporation
VBoxService.exe	0.14	1.916 K	5.064 K	640	VirtualBox Guest Additions S...	Oracle Corporation
svchost.exe	0.11	2.800 K	6.412 K	696	Processo host per servizi di ...	Microsoft Corporation
svchost.exe		14.140 K	13.416 K	776	Processo host per servizi di ...	Microsoft Corporation
audiodg.exe		15.468 K	15.240 K	976		
svchost.exe	0.01	4.760 K	11.848 K	844	Processo host per servizi di ...	Microsoft Corporation
dwm.exe		1.452 K	4.716 K	1852	Gestione finestre desktop	Microsoft Corporation
svchost.exe	0.02	9.300 K	20.524 K	888	Processo host per servizi di ...	Microsoft Corporation
svchost.exe		3.872 K	8.132 K	284	Processo host per servizi di ...	Microsoft Corporation
svchost.exe	< 0.01	5.084 K	9.608 K	1012	Processo host per servizi di ...	Microsoft Corporation
spoolsv.exe		1.500 K	3.964 K	1156	Applicazione sottosistema sp...	Microsoft Corporation
taskhost.exe	0.04	2.656 K	5.820 K	1216	Processo host per attività di ...	Microsoft Corporation
svchost.exe		9.416 K	11.476 K	1236	Processo host per servizi di ...	Microsoft Corporation
svchost.exe		3.856 K	7.568 K	1380	Processo host per servizi di ...	Microsoft Corporation
sppsvc.exe		2.472 K	8.080 K	1620	Servizio piattaforma protezio...	Microsoft Corporation
SearchIndexer.exe	0.51	19.152 K	12.292 K	1440	Microsoft Windows Search I...	Microsoft Corporation
SearchProtocolHost.exe	2.14	2.128 K	7.500 K	2492		
SearchFilterHost.exe	2.12	2.008 K	5.508 K	2512		
lsass.exe	0.22	2.848 K	8.592 K	480	Local Security Authority Proc...	Microsoft Corporation
lsam.exe	0.01	2.176 K	3.916 K	488		
csrss.exe	1.50	1.716 K	4.484 K	380		
conhost.exe	0.06	248 K	452 K	2476	Host finestra console	Microsoft Corporation
winlogon.exe		2.364 K	6.376 K	408		
explorer.exe	2.29	34.984 K	46.580 K	1876	Esplora risorse	Microsoft Corporation
VBoxTray.exe	0.04	2.040 K	6.388 K	1960	VirtualBox Guest Additions Tr...	Oracle Corporation
svchost.exe		2.124 K	7.488 K	100	Swinscowe Service Emulatio...	Swinscowe, www.swinsco...

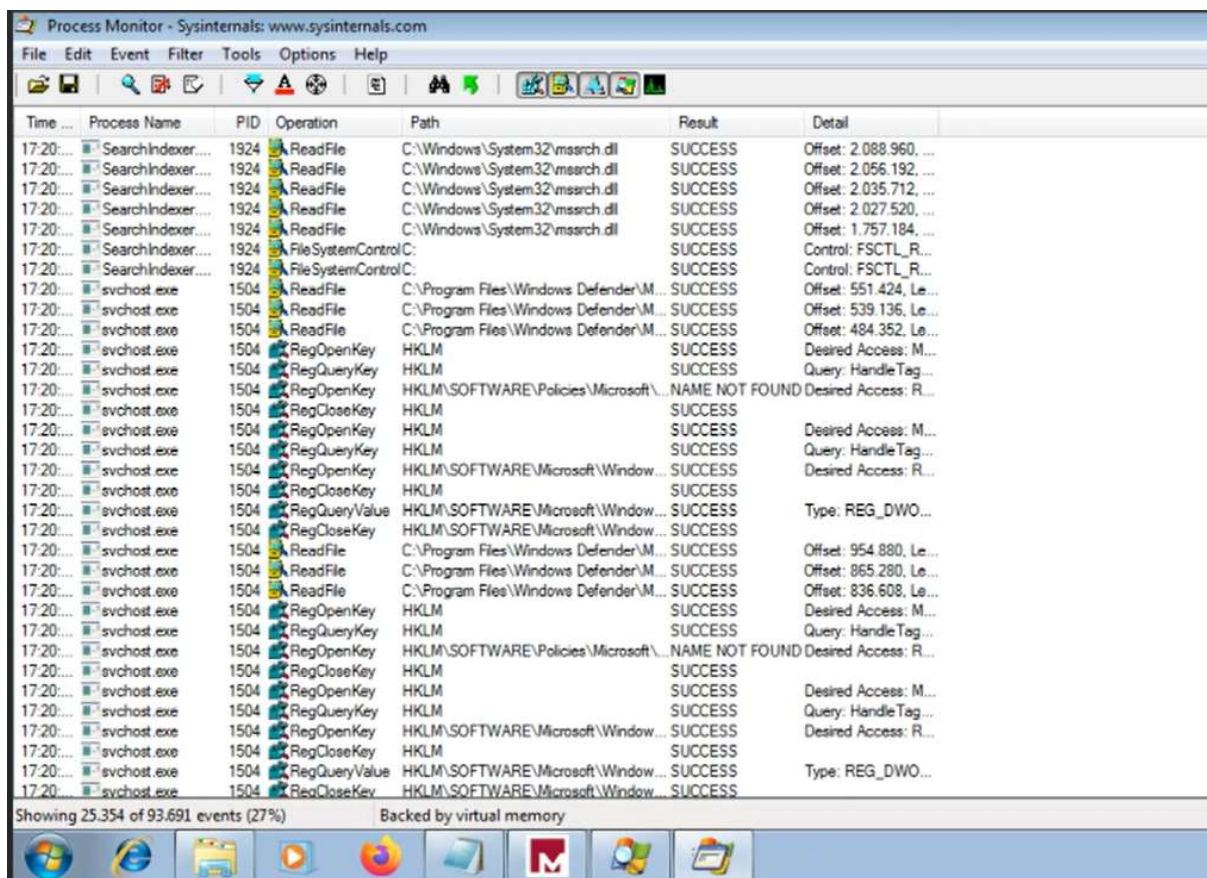
2. Attivazione del server DNS con ApateDNS



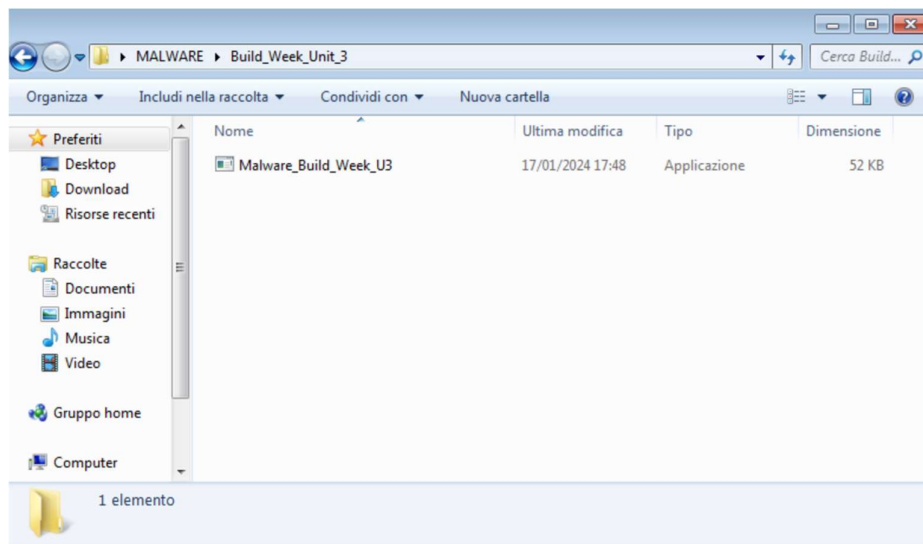
3. Salvataggio di un'istantanea del registro di sistema con RegShot



4. Avvio di ProcMon



5. Esecuzione del malware



Subito dopo l'esecuzione del malware, un elemento di particolare interesse emerge all'interno della sua cartella: la presenza del file msgina32.dll.



Questo file, solitamente presente nel sistema operativo Windows, svolge un ruolo importante nel processo di login e nella gestione del desktop. La sua presenza all'interno della cartella del malware solleva immediatamente alcuni dubbi e apre diverse piste investigative.

Possibili Implicazioni:

- **Comportamento Mimico:** Il malware potrebbe tentare di mascherarsi da un processo legittimo di Windows, sfruttando il nome "msgina32.dll" per ingannare gli utenti e i sistemi di sicurezza.

- ## 6. Arresto della cattura di ProcMon

File Edit Event Filter Tools Options Help

Process Monitor - Sysinternals: www.sysinternals.com

Time	Process	Operation	Path	Result	Detail
17:25	SearchIndexer.exe	File System Control	C:\	SUCCESS	Control: FSCTL_Q...
17:25	SearchIndexer.exe	File System Control	C:\	SUCCESS	Control: FSCTL_R...
17:25	SearchIndexer.exe	File System Control	C:\	SUCCESS	Control: FSCTL_R...
17:25	VBoxService.exe	Thread Create		SUCCESS	Thread ID: 2352
17:25	svchost.exe	RegOpenKey	HKLM	SUCCESS	Desired Access: M...
17:25	svchost.exe	RegQueryValue	HKLM	SUCCESS	Query: HandleTag...
17:25	svchost.exe	RegOpenKey	HKLM\SYSTEM\CurrentControlSet\Ser...	REPARSE	Desired Access: R...
17:25	svchost.exe	RegOpenKey	HKLM\System\CurrentControlSet\Servi...	NAME NOT FOUND	Desired Access: R...
17:25	svchost.exe	RegCloseKey	HKLM	SUCCESS	
17:25	svchost.exe	RegOpenKey	HKLM	SUCCESS	Desired Access: M...
17:25	svchost.exe	RegQueryValue	HKLM	SUCCESS	Query: HandleTag...
17:25	svchost.exe	RegOpenKey	HKLM\SOFTWARE\Microsoft\WBEM\...	NAME NOT FOUND	Desired Access: R...
17:25	svchost.exe	RegCloseKey	HKLM	SUCCESS	
17:25	lsass.exe	RegOpenKey	HKLM\SAM\SAM\DOMAINS\Account\...	NAME NOT FOUND	Desired Access: R...
17:25	lsass.exe	RegOpenKey	HKLM\SAM\SAM\DOMAINS\Account\...	NAME NOT FOUND	Desired Access: R...
17:25	lsass.exe	RegOpenKey	HKLM\SAM\SAM\DOMAINS\Account\...	SUCCESS	Desired Access: R...
17:25	lsass.exe	RegQueryValue	HKLM\SAM\SAM\DOMAINS\Account\...	SUCCESS	Type: REG_BINA...
17:25	lsass.exe	RegCloseKey	HKLM\SAM\SAM\DOMAINS\Account\...	SUCCESS	
17:25	lsass.exe	RegOpenKey	HKLM\SAM\SAM\DOMAINS\Account\...	NAME NOT FOUND	Desired Access: R...
17:25	lsass.exe	RegOpenKey	HKLM\SAM\SAM\DOMAINS\Account\...	NAME NOT FOUND	Desired Access: R...

7. Stop del server ApateDNS

The screenshot shows the ApatеDNS application interface. At the top, there's a title bar "ApatеDNS" with standard window controls. Below it are two tabs: "Capture Window" and "DNS Hex View". The main area displays a log of DNS transactions.

Time	Domain Requested	DNS Returned
17:20:25	255.1.168.192.in-addr.arpa	FOUND
17:22:17	3.0.0.0.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.2.0.f.f.i.p.6.a.r.p.a	FOUND
17:22:17	252.0.0.224.in-addr.arpa	FOUND
17:22:20	spynet2.microsoft.com	FOUND
17:22:20	spynet2.microsoft.com	FOUND
17:24:01	c.0.2.0.f.f.i.p.6.a.r.p.a	FOUND
17:24:01	c.0.2.0.f.f.i.p.6.a.r.p.a	FOUND
17:24:51	2.0.0.0.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.2.0.f.f.i.p.6.a.r.p.a	FOUND
17:32:22	spynet2.microsoft.com	FOUND
17:32:22	spynet2.microsoft.com	FOUND

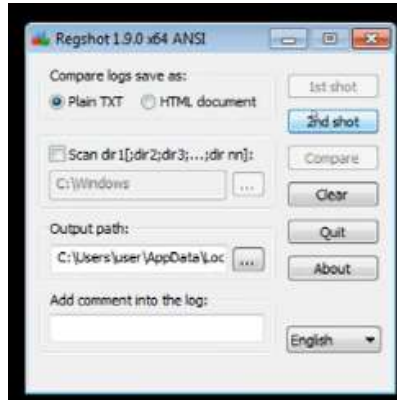
Below the table, there's a log window showing the following messages:

```
[+] Using 192.168.1.50 as return DNS IP!
[+] DNS set to 127.0.0.1 on Scheda desktop Intel(R) PRO/1000 MT.
[+] Sending valid DNS response of first request.
[+] Server started at 17:16:33 successfully.
[+] Stopping Server...
[+] Static DNS detected, setting back DNS to(192.168.1.50 ).
[+] DNS Restored.
[+] Interfaces list has been refreshed.
```

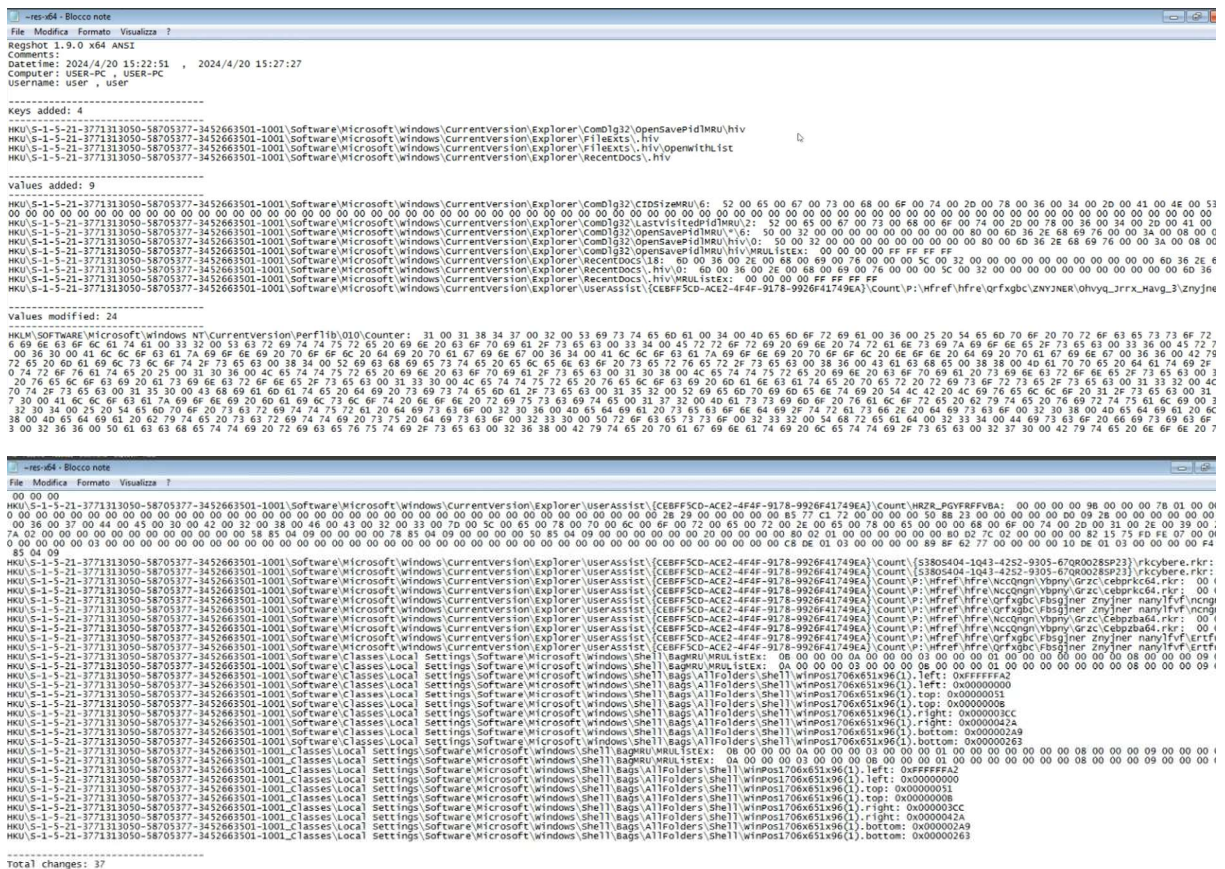
At the bottom, there are configuration fields and control buttons:

- DNS Reply IP (Default: Current Gateway/DNS):** 192.168.1.50
- # of NXDOMAIN's:** 0
- Selected Interface:** Scheda desktop Intel(R) PRO/1000 MT
- Start Server** button
- Stop Server** button

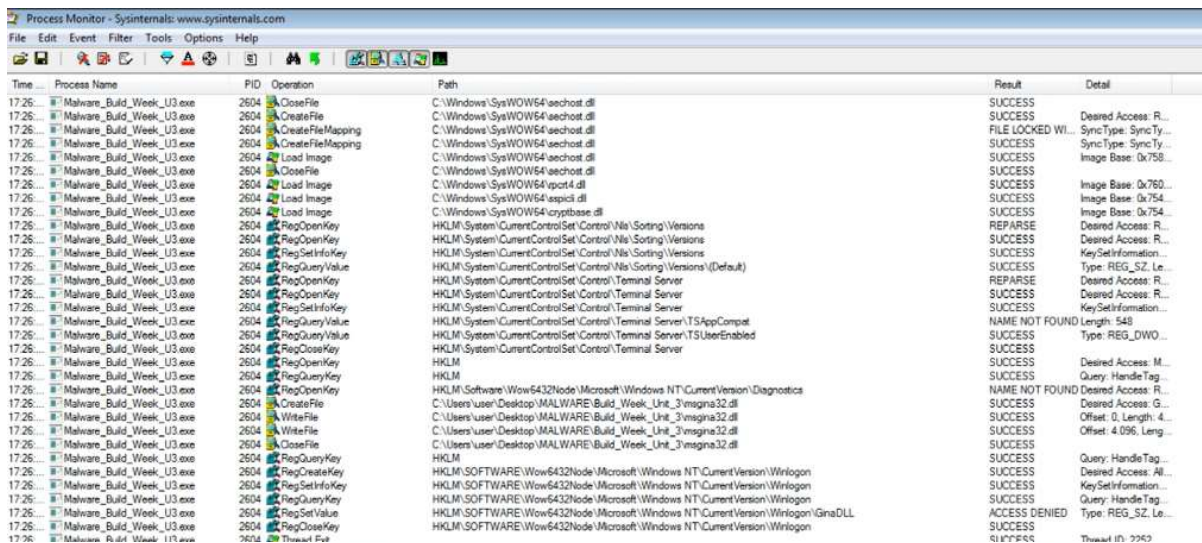
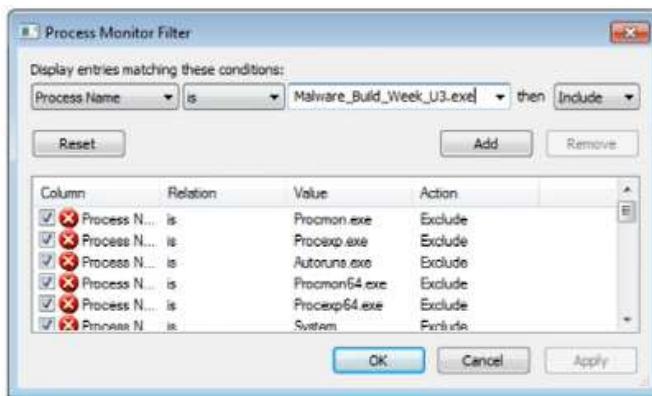
8. Salvataggio di una seconda istantanea del registro di sistema con RegShot:



9. Confronto delle due istantanee di RegShot

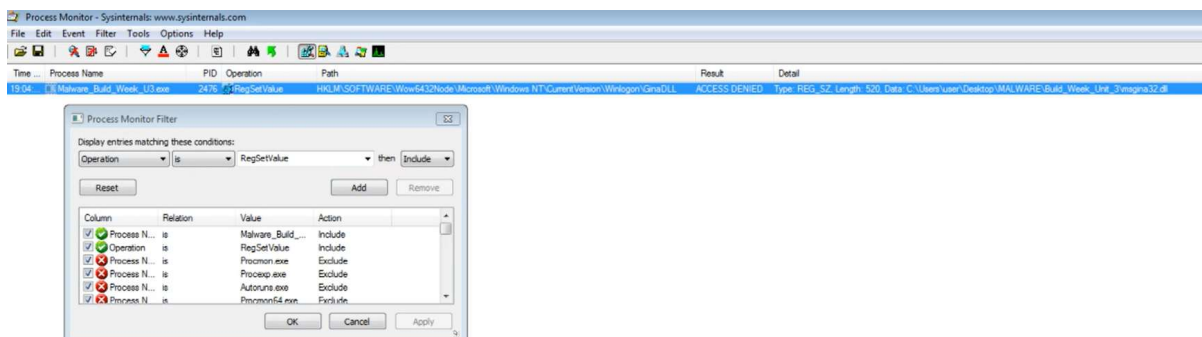


10. Creazione filtro in ProcMon per analizzare solo i processi del malware



L'esecuzione del malware ha generato un'attività sospetta nel registro di sistema, suggerendo la creazione di chiavi e valori potenzialmente dannosi. Per approfondire questa scoperta, è necessario un'analisi attenta delle modifiche apportate al registro.

VIEW REGISTRO CON FILTRO CREAZIONE CHIAVI



L'analisi del registro di sistema ha portato alla luce una modifica significativa: la chiave HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Winlogon\Gina DLL ha subito un'alterazione del suo valore REG_SZ. Questa scoperta richiede un'attenzione immediata per comprendere le implicazioni e il potenziale impatto del malware sul sistema.

Interpretazione della Modifica:

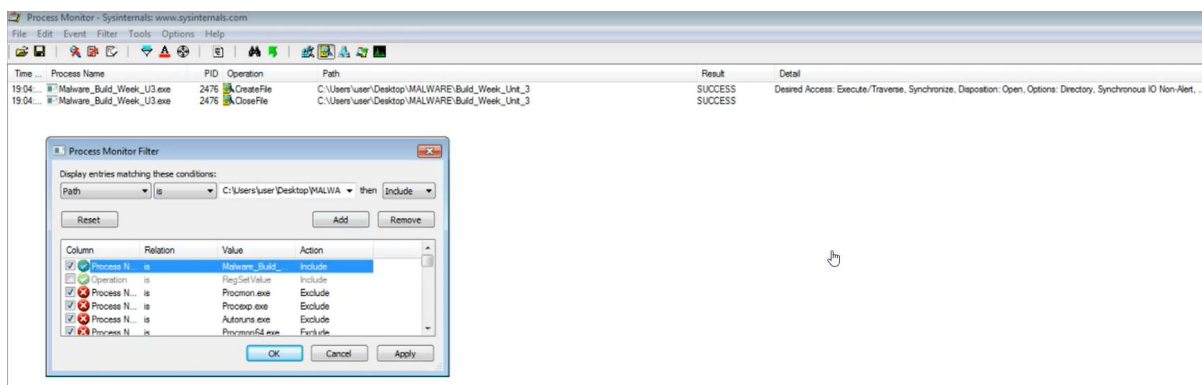
- **Gina DLL:** La chiave Gina DLL all'interno del registro di sistema regola il gestore di accesso grafico di Windows, il programma responsabile del processo di login e della gestione del desktop.
- **Valore REG_SZ:** Il valore REG_SZ associato a questa chiave specifica il nome del file DLL da utilizzare come gestore di accesso grafico.

Possibili Implicazioni:

- **Sostituzione del Gestore di Accesso:** Il malware potrebbe aver modificato il valore REG_SZ per sostituire il gestore di accesso grafico legittimo con uno dannoso, al fine di intercettare le credenziali di accesso degli utenti o manipolare l'ambiente desktop.
- **Persistenza del Malware:** La modifica del valore REG_SZ potrebbe essere un tentativo del malware di garantire la sua persistenza nel sistema, assicurando che venga caricato automaticamente al login di Windows.
- **Comportamento Dannoso:** Il gestore di accesso grafico dannoso potrebbe essere utilizzato per eseguire diverse azioni malevole, come l'installazione di software indesiderato, il furto di dati sensibili o il monitoraggio delle attività degli utenti.

VIEW FILESYSTEM CON FILTRO PATH CARTELLA MALWARE

L'esecuzione del malware ha portato alla creazione di un file DLL sospetto all'interno della sua cartella. Per approfondire questa scoperta, è necessario un'analisi approfondita delle attività del file system che hanno generato questa modifica, applicando un filtro sul path della cartella del malware per monitorare esclusivamente le attività all'interno di quella directory.



L'analisi del file system ha rivelato un'attività sospetta associata al processo con PID 2476: l'esecuzione delle operazioni CreateFile e CloseFile.

ANALISI STATICA AVANZATA

L'analisi statica avanzata rappresenta un potente strumento per penetrare nelle profondità del codice malware, svelando i suoi meccanismi interni, le sue intenzioni e le sue potenziali minacce. Nel mondo dell'analisi dei software, i disassembler svolgono un ruolo fondamentale nel tradurre il codice binario in un linguaggio di programmazione comprensibile, permettendo agli analisti di comprendere il funzionamento interno di programmi complessi, in particolare quelli dannosi.

DISASSEMBLER

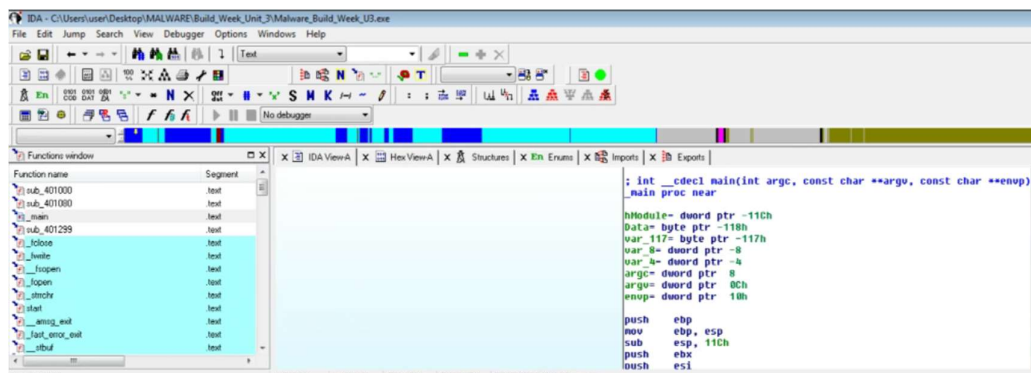
I disassembler sono strumenti software progettati per invertire il processo di assemblaggio, ovvero la conversione del codice sorgente in linguaggio di programmazione in codice binario eseguibile da un computer. Il processo di disassemblaggio scompone il codice binario in istruzioni mnemoniche, che rappresentano le singole operazioni che la CPU può eseguire. In questo modo, i disassembler forniscono una rappresentazione strutturata del codice binario, rendendolo più accessibile e analizzabile.

IDA PRO

IDA Pro (Interactive Disassembler Pro) è un disassembler leader nel settore, rinomato per la sua capacità di analizzare codice binario complesso e fornire una visione dettagliata del comportamento del software. Con le sue potenti funzionalità e la sua interfaccia intuitiva, IDA Pro permette agli analisti di:

- **Caricare e disassemblare** file eseguibili e librerie in vari formati binari.
- **Visualizzare il codice sorgente** disassemblato in modo strutturato e comprensibile.
- **Identificare funzioni**, variabili e strutture di dati all'interno del codice.
- **Tracciare il flusso di controllo** del programma e analizzare le sue condizioni e i suoi loop.
- **Annotare il codice** con commenti e note per organizzare l'analisi e condividere le informazioni con altri analisti.

Per comprendere appieno il comportamento di un programma, è fondamentale analizzare la sua funzione principale, ovvero il punto di ingresso da cui ha inizio l'esecuzione. In questa analisi, ci concentreremo sulla funzione main del programma in esame, esaminando i suoi parametri di input e le variabili locali dichiarate al suo interno.



PARAMETRI INPUT FUNZIONE MAIN

La funzione **main** riceve tre parametri di input standard:

1. **argc (int):** Un intero che rappresenta il numero di argomenti passati al programma dalla riga di comando.
2. ****argv (const char):** Un array di puntatori a stringhe che contiene gli argomenti passati al programma dalla riga di comando. L'elemento `argv[0]` contiene il nome del programma stesso, mentre gli elementi successivi contengono gli argomenti aggiuntivi.
3. ****envp (const char):** Un array di puntatori a stringhe che contiene le variabili d'ambiente del sistema operativo. Le variabili d'ambiente forniscono informazioni sulla configurazione del sistema e sulle impostazioni utente.

IDENTIFICAZIONE VARIABILI LOCALI

Nell'analisi del codice assembly, le variabili locali possono essere identificate utilizzando offset negativi. Questa tecnica si basa sul fatto che le variabili locali vengono memorizzate nello stack di esecuzione, che è una struttura di dati LIFO (Last In, First Out). In altre parole, le variabili più recenti vengono memorizzate nella parte superiore dello stack, mentre le variabili più vecchie vengono memorizzate in posizioni con offset negativi rispetto al puntatore del frame della funzione. Dall'analisi del codice possiamo identificare 5 variabili:

1. hModule (dword ptr -11ch):

Tipo: dword (indica che la variabile memorizza un valore a 32 bit, spesso utilizzato per indirizzi di memoria o interi a 32 bit).

Offset: -11ch (equivalente a -188 in decimale). Questo offset negativo indica che la variabile hModule è memorizzata a 188 byte sotto il puntatore del frame della funzione.

Possibile utilizzo: Potrebbe contenere l'handle di un modulo caricato dinamicamente (DLL) utilizzando la funzione LoadLibrary.

2. Data (byte ptr -118h):

Tipo: byte (indica che la variabile memorizza un singolo byte di dati).

Offset: -118h (equivalente a -184 in decimale). Questo offset è vicino a hModule, suggerendo che potrebbero essere memorizzati in posizioni adiacenti nello stack.

Possibile utilizzo: Potrebbe contenere un singolo byte di dati utilizzato per un'operazione specifica all'interno della funzione.

3. Var_117 (byte ptr -117h):

Tipo: byte (indica che la variabile memorizza un singolo byte di dati).

Offset: -117h (equivalente a -183 in decimale). Ancora più vicino a hModule e Data, rafforzando l'idea che siano allocati consecutivamente nello stack.

Possibile utilizzo: Potrebbe essere un altro singolo byte di dati correlato a hModule o Data.

4. Var_8 (dword ptr -8):

Tipo: dword (indica che la variabile memorizza un valore a 32 bit).

Offset: -8 (equivalente a -8 in decimale). Questo offset è relativamente piccolo, indicando che Var_8 è memorizzata vicino alla parte superiore dello stack per la funzione corrente.

Possibile utilizzo: Potrebbe contenere un valore intermedio o un risultato calcolato utilizzato all'interno della funzione.

5. Var_4 (dword ptr -4):

Tipo: dword (indica che la variabile memorizza un valore a 32 bit).

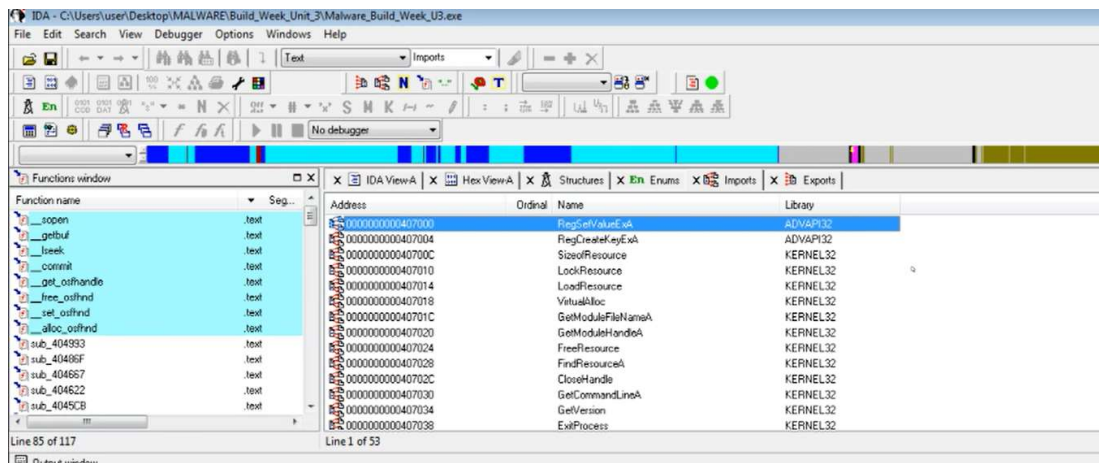
Offset: -4 (equivalente a -4 in decimale). Posizione ancora più vicina alla parte superiore dello stack rispetto a Var_8.

Possibile utilizzo: Potrebbe essere un argomento passato alla funzione main o una variabile temporanea utilizzata per un breve periodo all'interno della funzione.

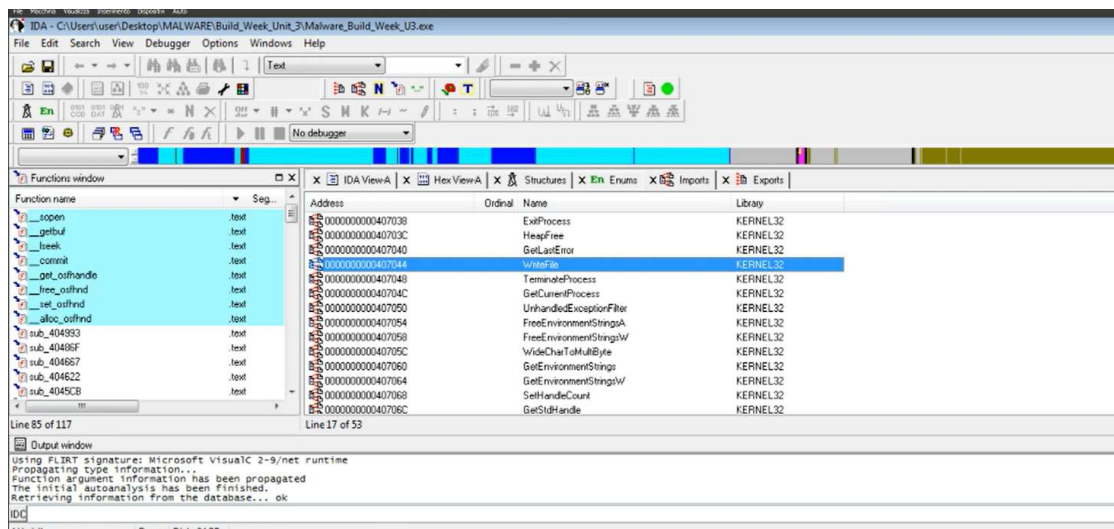
ANALISI IMPORT E IDENTIFICAZIONE MALWARE

1. ADVAPI32

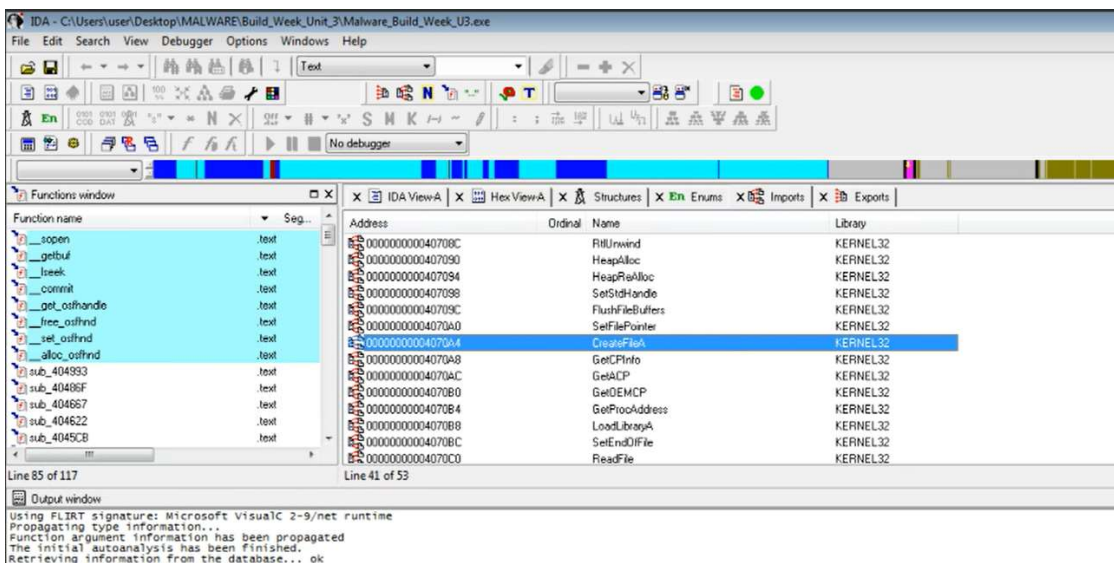
- FindResourceA
- LoadResource



2. KERNEL32 (WriteFile)



3. KERNEL32(CreateFile)



L'analisi degli import di un malware rappresenta un passaggio fondamentale per comprenderne le funzionalità e il modus operandi. In questo caso, l'esame delle librerie importate advapi32 e kernel32 e delle specifiche funzioni RegSetValueExA, FindResourceA, LoadResource, CreateFile e WriteFile suggerisce la possibile presenza di un dropper malware.

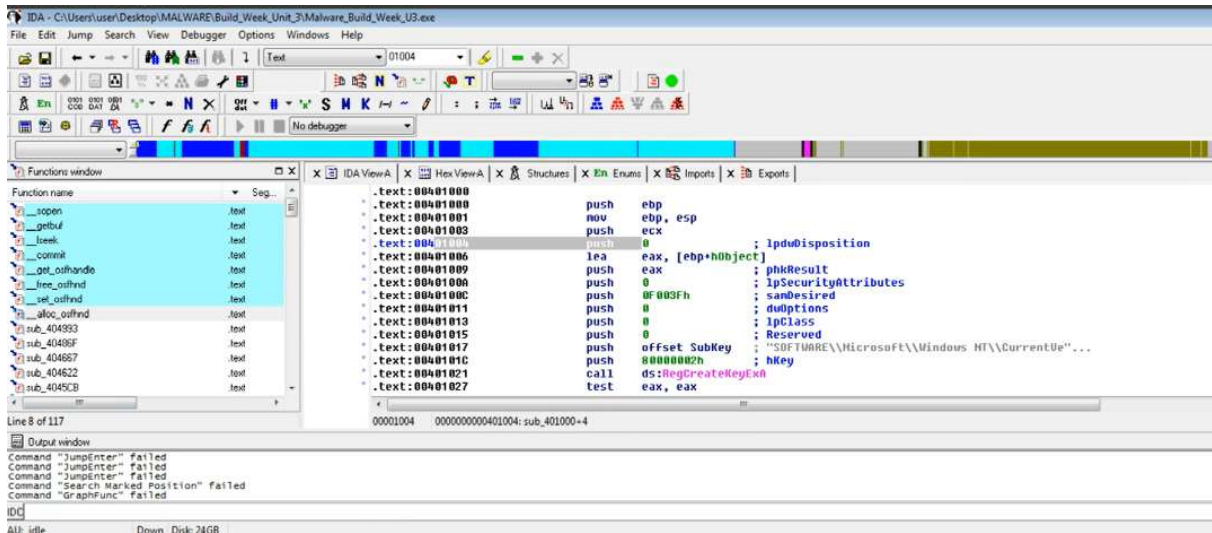
EVIDENZE A FAVORE DELL'IPOTESI DROPPER

- **Importazione di Funzioni per la Gestione delle Risorse:** Le funzioni FindResourceA e LoadResource sono tipicamente utilizzate per accedere alle risorse interne di un file eseguibile, come ad esempio un'immagine o un dato binario. Nel contesto di un dropper, queste funzioni potrebbero essere impiegate per estrarre il payload dannoso nascosto all'interno delle risorse del file.
- **Importazione di Funzioni per la Scrittura su File:** Le funzioni CreateFile e WriteFile permettono la creazione e la scrittura di file sul sistema. Un dropper potrebbe utilizzare queste funzioni per estrarre il payload dannoso dalle risorse e salvarlo come nuovo file eseguibile sul disco.

ESCLUSIONE DI ALTRE IPOTESI

- **Assenza di API per il Download di File:** La mancata importazione dell'API URLDownloadToFile, comunemente utilizzata dai downloader, esclude la possibilità che il malware sia un downloader diretto.
- **Assenza di API per il Monitoraggio degli Eventi:** L'assenza dell'API SetWindowsHookEx, impiegata per il monitoraggio degli eventi delle periferiche, suggerisce che il malware non sia un keylogger.
- **Assenza di Libreria Winsock:** La mancata importazione della libreria Winsock, necessaria per le funzionalità di rete, esclude la possibilità che il malware sia una backdoor.

ANALISI APPROFONDIRITA DELLA CHIAMATA A REGCREATEKEYEXA



Nell'analisi del codice assembly, l'esame di chiamate a funzioni specifiche può fornire informazioni preziose sul comportamento del programma. In questo caso, ci concentriamo sulla chiamata alla funzione RegCreateKeyExA all'indirizzo di memoria 00401021.

SPIEGAZIONE DELLA FUNZIONE REGCREATEKEYEXA

La funzione RegCreateKeyExA è utilizzata per creare una nuova chiave nel registro di sistema di Windows. Essa permette di specificare vari attributi della chiave, come il suo nome, le autorizzazioni di accesso e il tipo di chiave. La funzione restituisce un handle alla nuova chiave creata, che può essere utilizzato per ulteriori operazioni di registro.

PARAMETRI PASSATI TRAMITE LO STACK

Nell'analisi del codice assembly, possiamo osservare che i parametri della funzione RegCreateKeyExA vengono passati tramite lo stack. Questo è evidente dalle istruzioni push che precedono la chiamata alla funzione. In totale, possiamo identificare 9 parametri:

1. **lpdwDisposition**: Un puntatore a una variabile DWORD che riceverà la disposizione della chiave esistente se presente.
2. **phkresult**: Un puntatore a una variabile HANDLE che riceverà il handle alla nuova chiave creata.
3. **lpsecurityattributes**: Un puntatore a una struttura SECURITY_ATTRIBUTES che specifica le autorizzazioni di accesso per la chiave.

4. **sandesired**: Un valore DWORD che specifica il valore desiderato per la chiave.
5. **dwoptions**: Un valore DWORD che specifica le opzioni di creazione della chiave.
6. **Ipclass**: Un puntatore a una stringa che specifica la classe della chiave.
7. **reserved**: Un valore DWORD riservato per uso futuro.
8. **subkey**: Un puntatore a una stringa che specifica il nome della sottochiave da creare.
9. **hkey**: Un valore HKEY che specifica la chiave padre in cui creare la nuova sottochiave.

PUSH PARAMETRO "SUBKEY" INDIRIZZO 00401017

Il parametro subkey rappresenta il nome della sottochiave che si desidera creare all'interno della chiave padre specificata dal parametro hkey. Il valore della stringa subkey viene memorizzato in memoria e passato alla funzione RegCreateKeyExA tramite lo stack.

ANALISI CODICE ASSEMBLY

Nell'analisi del codice assembly, le sequenze di istruzioni possono rivelare il flusso di controllo del programma e le condizioni che influenzano l'esecuzione. In questo caso, esaminiamo le istruzioni comprese tra gli indirizzi 00401027 e 00401029:

```
.text:00401027 push    eax
.text:00401028 push    ebx
.text:00401029 push    ecx
.text:0040102A push    edx
.text:0040102B push    esi
.text:0040102C push    offset SubKey
.text:0040102D push    8000002h
.text:0040102E call    ds:RegCreateKeyExA
.text:0040102F jz      short loc_401032
.text:00401030 mov     eax, 1
.text:00401031 jnp     short loc_401028
.text:00401032 loc_401032: ; CODE XREF: sub_401000+29j
```

Line 8 of 117

Output window

Command "JumpEnter" failed
Command "Search Marked Position" failed
Command "GraphFunc" failed
Retrieving information from the database... ok
Retrieving information from the database... ok

IDA
AU: idle Down: Disk: 24GB

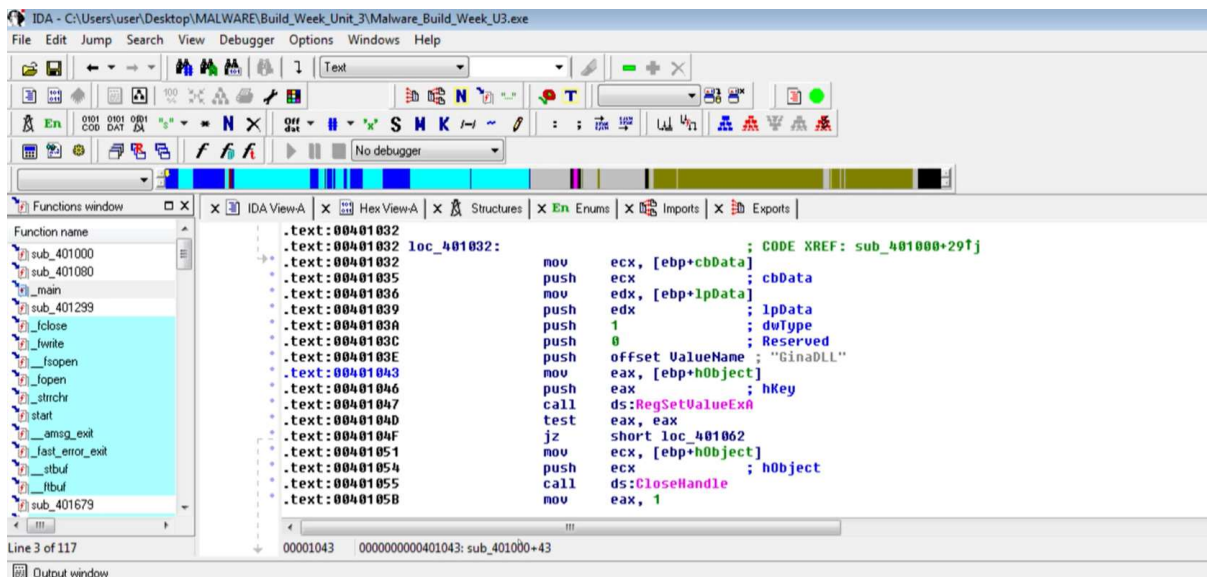
L'istruzione `test eax, eax` confronta il valore del registro EAX con se stesso. In altre parole, verifica se EAX è uguale a zero. Se EAX è uguale a zero, il flag di zero (ZF) viene impostato su 1. Se EAX non è uguale a zero, il flag di zero viene impostato su 0.

L'istruzione `jz short_loc_401032` rappresenta un salto condizionale che controlla il flag di zero (ZF). Se il flag di zero è impostato su 1 (indicando che EAX è uguale a zero), il programma salta all'istruzione situata all'indirizzo 00401032. Se il flag di zero è impostato su 0 (indicando che EAX non è uguale a zero), il programma continua l'esecuzione dell'istruzione successiva alla `jz`.

INTERPRETAZIONE COME COSTRUTTO IF-ELSE

In base alla combinazione di `test eax, eax` e `jz short_loc_401032`, possiamo interpretare questa sequenza di istruzioni come un costrutto if-else. Se il valore di EAX è zero, la condizione if è vera e il programma salta all'istruzione 00401032. Se il valore di EAX è diverso da zero, la condizione if è falsa e il programma continua l'esecuzione dell'istruzione successiva alla `jz`.

ANALISI DELLA CHIAMATA ALLA FUNZIONE REGSETVALUEEXA ALL'INDIRIZZO 00401047



```
.text:00401032
.text:00401032 loc_401032:
.text:00401035
.text:00401036
.text:00401039
.text:0040103A
.text:0040103C
.text:0040103E
.text:00401043
.text:00401046
.text:00401047
.text:0040104D
.text:0040104F
.text:00401051
.text:00401054
.text:00401055
.text:0040105B

mov     ecx, [ebp+cbData] ; CODE XREF: sub_401000+291j
push    ecx               ; cbData
mov     edx, [ebp+lpData]
push    edx               ; lpData
push    1                 ; dwType
push    0                 ; Reserved
push    offset ValueName ; "GinaDLL"
mov     eax, [ebp+hObject]
push    eax               ; hKey
call    ds:RegSetValueExA
test    eax, eax
jz      short loc_401062
mov     ecx, [ebp+hObject]
push    ecx               ; hObject
call    ds:CloseHandle
mov     eax, 1
```

Nell'esame del codice assembly, l'identificazione di chiamate a funzioni specifiche è fondamentale per comprendere il comportamento del programma. In questo caso, ci concentriamo sulla chiamata alla funzione `RegSetValueExA` all'indirizzo 00401047.

DESCRIZIONE DELLA FUNZIONE REGSETVALUEEXA

La funzione RegSetValueExA viene utilizzata per impostare un valore in una chiave del registro di sistema di Windows. Permette di specificare vari attributi del valore, come il suo nome, il tipo di dato, i dati da impostare e le autorizzazioni di accesso. La funzione non restituisce alcun valore.

COMMENTO IDAPRO PER IL PUSH OFFSET VALUENAME

La presenza di un commento IDAPro associato al push dell'offset ValueName fornisce informazioni preziose sul valore che verrà impostato nella chiave di registro. Il commento "GinaDLL" suggerisce che il valore da impostare è la stringa "GinaDLL".

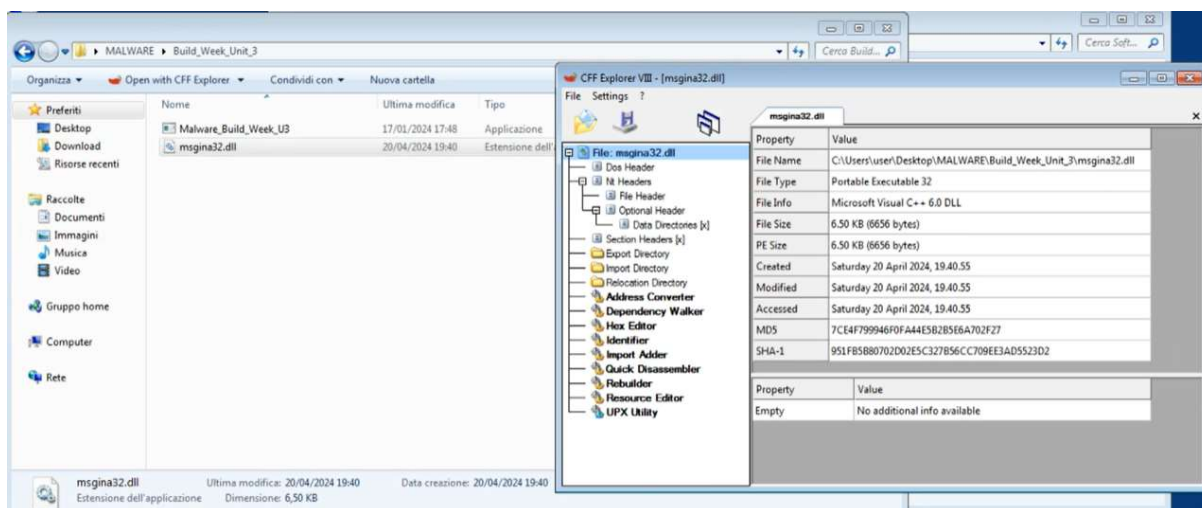
ANALISI DELLA DLL ESTRATTA

Oltre all'analisi del file principale, abbiamo analizzato anche la DLL generata dal file stesso. L'esame di questa DLL può fornire informazioni preziose sul comportamento del malware e sulle sue potenziali minacce.

PROCEDIMENTI DI ANALISI

L'analisi della DLL ha comportato i seguenti passaggi:

1. Recupero hash tramite CFF Explorer



2. Analisi con VirusTotal

The screenshot shows the VirusTotal analysis page for a file named 'magna32.dll'. The file's SHA-256 hash is 'fba4f61bccd5bab1cad0ab9e57f6f3092a8bd4dd30adfc4853e89ba96afc93f9'. The file size is 6.50 KB and it was last modified 19 days ago. A community score of 51/71 is displayed, indicating that 51 out of 71 security vendors and sandboxes flagged the file as malicious. The file is categorized as 'trojan.fragtor.tiggre'. The 'Security vendors' analysis table shows detections from various vendors, including Alibaba, ALYac, Arcabit, AVG, BitDefender, Bkav Pro, CrowStrike Falcon, Cynet, DrWeb, and others. The table also lists the specific threat names detected by each vendor.

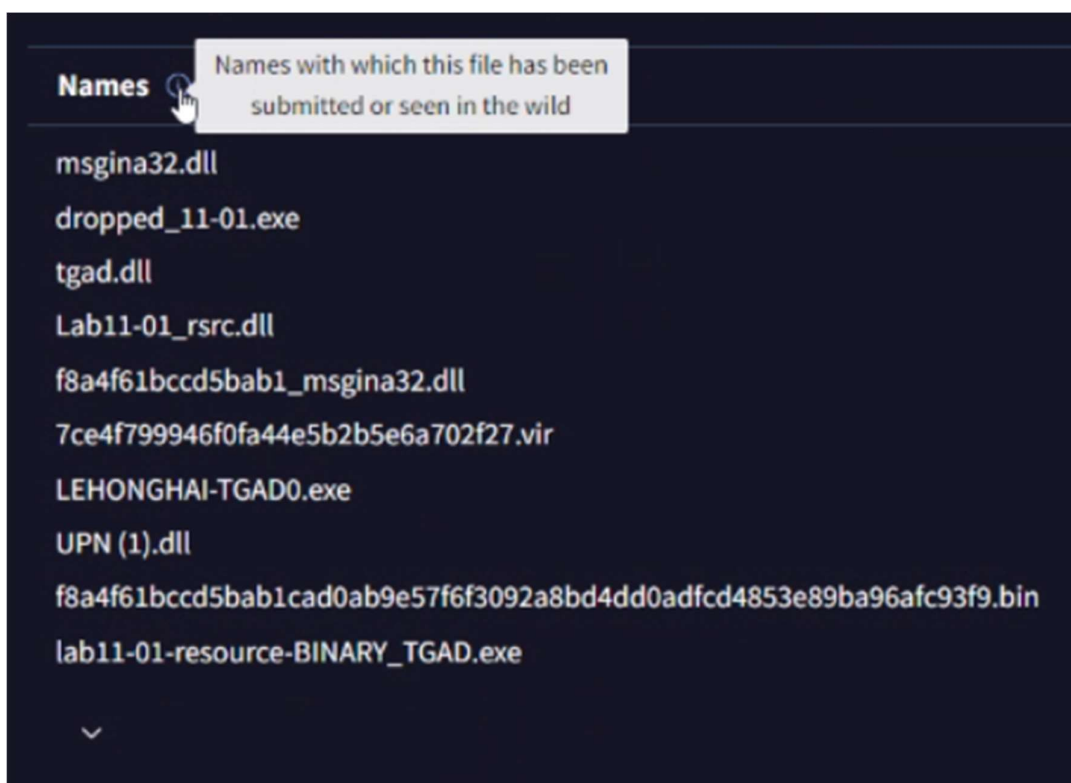
Security vendors' analysis	Threat categories	Family labels
Alibaba	Trojan.Win32/Tiggre.3876ba16	AICloud
ALYac	Gen.Variant.Fragtor.510142	Antiy-AVL
Arcabit	Trojan.Fragtor.D/CBBE	Avast
AVG	Win32-Trojan-gen	Avira (no cloud)
BitDefender	Gen.Variant.Fragtor.510142	BitDefender/Theta
Bkav Pro	W32.Common.146/CBBE	ClamAV
CrowStrike Falcon	Win/malicious_confidence_100% (W)	Cylance
Cynet	Malicious (score: 100)	DeepInstinct
DrWeb	Backdoor.Siggen2.1689	Emisoft

3. Sezione MITRE ATT&CK

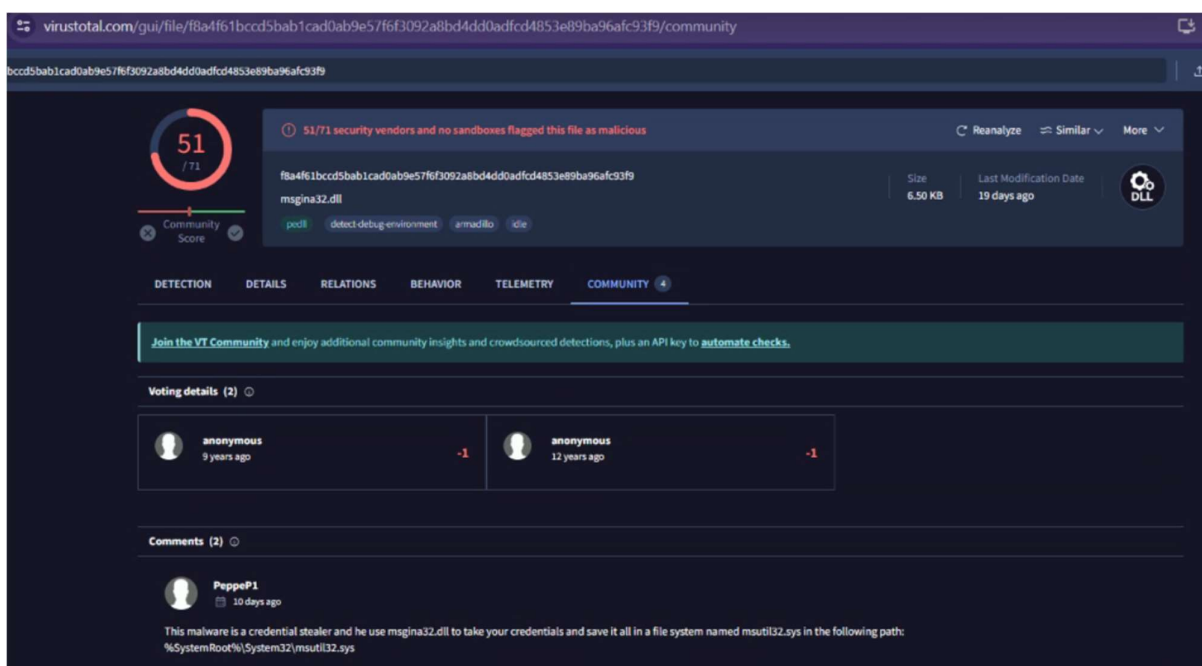
The screenshot shows the MITRE ATT&CK Tactics and Techniques page. The page lists various tactics and techniques, including Execution, Persistence, Privilege Escalation, Defense Evasion, and Discovery. Each tactic is associated with a specific technique ID and a brief description of the technique.

- Execution** (T1002)
 - Shared Modules (T1129): Link function at runtime on Windows
- Persistence** (T1003)
 - Event Triggered Execution (T1546): Persist via GinaDLL registry key
 - DLL Side-Loading (T1574.002): Tries to load missing DLLs
- Privilege Escalation** (T1004)
 - Event Triggered Execution (T1546): Persist via GinaDLL registry key
 - DLL Side-Loading (T1574.002): Tries to load missing DLLs
- Defense Evasion** (T1005)
 - Rundll32 (T1135.011): Runs a DLL by calling functions
 - Virtualization/Sandbox Evasion (T1407): Checks if the current process is being debugged
 - DLL Side-Loading (T1574.002): Tries to load missing DLLs
- Discovery** (T1007)
 - System Information Discovery (T1082): Reads software policies
 - File and Directory Discovery (T1083): Get common file path
 - Virtualization/Sandbox Evasion (T1407): Checks if the current process is being debugged

4. Sezione Names



5. Sezione Community



ANALISI DINAMICA AVANZATA

L'analisi dinamica avanzata va oltre l'osservazione del comportamento superficiale del software durante l'esecuzione. Questa tecnica coinvolge l'uso di strumenti avanzati come debugger e analizzatori di traffico di rete per ottenere una comprensione più approfondita delle attività del programma. Gli analisti di sicurezza utilizzano l'analisi dinamica avanzata per tracciare le chiamate di sistema, monitorare la memoria e identificare eventuali tecniche di evasione utilizzate dal software per eludere la rilevazione. Questo tipo di analisi richiede competenze più avanzate nel campo della sicurezza informatica e strumenti specializzati per rilevare e analizzare minacce complesse.

DEBUGGER E OLLYDBG

I debugger sono strumenti software che consentono di monitorare e controllare l'esecuzione di un programma. Permettono di impostare breakpoint, visualizzare lo stato delle variabili e della memoria, eseguire il programma passo dopo passo e modificare il flusso di esecuzione. OllyDbg è un debugger noto per la sua semplicità d'uso e le sue funzionalità avanzate per l'analisi di malware.

APPLICAZIONE ALL'ANALISI DINAMICA AVANZATA

L'analisi dinamica avanzata sfrutta i debugger per osservare in dettaglio il comportamento di un programma sospetto in un ambiente controllato.

Questo tipo di analisi può rivelare informazioni preziose che potrebbero non essere evidenti dall'analisi statica, come:

1. **Comportamento in Esecuzione:** Come il programma interagisce con il sistema operativo, i file di sistema e altri programmi.
2. **Caricamento di Moduli:** Identificazione dei moduli DLL e altri eseguibili caricati dal programma.
3. **Attività di Rete:** Comunicazione del programma con server esterni e scambio di dati.
4. **Tecniche di Evasione:** Modalità con cui il programma tenta di eludere i sistemi di sicurezza e l'analisi statica.

ANALISI APPROFONDATA E IDENTIFICAZIONE DEL MALWARE

Attraverso un'analisi completa che ha incluso sia l'analisi statica (basica-avanzata) che l'analisi dinamica avanzata, siamo stati in grado di determinare con certezza che il malware in esame è un dropper. Un dropper è un tipo di malware progettato per scaricare e installare altri malware sul sistema compromesso.

INFORMAZIONI ESTRATTE

L'analisi ha permesso di estrarre diverse informazioni preziose sul dropper, tra cui:

- **Meccanismi di Dropping:** Come il dropper scarica e installa il payload dannoso sul sistema.
- **Payload Dannoso:** Il tipo di malware che il dropper scarica e installa.
- **Comportamenti Secondari:** Eventuali altre azioni dannose eseguite dal dropper oltre al dropping del payload.
- **Comunicazioni di Rete:** Le comunicazioni del dropper con server esterni o con altri sistemi compromessi.
- **Tecniche di Evasione:** Le tecniche utilizzate dal dropper per ostacolare l'analisi e la detection da parte degli antivirus e degli strumenti di sicurezza.

IMPORTANZA DELL'ANALISI MALWARE

L'analisi approfondita di questo malware ha dimostrato ancora una volta l'importanza di un'analisi completa e rigorosa per comprendere appieno le minacce poste dai malware. L'analisi ci ha permesso di identificare la natura del dropper, il suo payload dannoso e i suoi comportamenti, informazioni cruciali per la mitigazione e la risposta agli incidenti.

OMISSIONE DELL'ANALISI DINAMICA AVANZATA CON OLLYDBG

Alla luce delle informazioni complete ottenute attraverso l'analisi statica e le tecniche di analisi dinamica avanzata impiegate, si ritiene che l'analisi in debug tramite OllyDbg non sia necessaria in questo caso. Le informazioni già raccolte forniscono una comprensione approfondita del malware e del suo comportamento, rendendo superflua l'analisi in debug per questo specifico scenario.

RACCOMANDAZIONI

Sulla base dei risultati dell'analisi, si raccomandano le seguenti azioni:

- **Rimozione del Malware:** Implementare misure adeguate per rimuovere il dropper e il suo payload dal sistema compromesso.
- **Prevenzione di Future Infezioni:** Aggiornare i sistemi e le applicazioni con le ultime patch di sicurezza, implementare soluzioni antivirus e anti-malware efficaci e sensibilizzare gli utenti sulle buone pratiche di sicurezza informatica.
- **Monitoraggio Continuo:** Monitorare il sistema per rilevare eventuali attività sospette o indicazioni di una nuova infezione da malware.
- **Condivisione delle Informazioni:** Condividere le informazioni estratte dal malware con le autorità competenti e con la community di sicurezza informatica per contribuire a migliorare la detection e la prevenzione di future minacce.

CONCLUSIONE

L'analisi condotta ha permesso di comprendere a fondo la natura e il comportamento del malware in esame, fornendo informazioni preziose per la sua rimozione e la prevenzione di future infezioni. L'omissione dell'analisi in debug con OllyDbg, in questo caso specifico, è giustificata dalla completezza delle informazioni già ottenute. L'analisi malware rimane un processo fondamentale per la sicurezza informatica e la sua importanza è destinata a crescere con l'evolversi delle minacce informatiche.