PREPARED BY
PHANTOM SRL

29/03/2024

REPORT S10-L5

ALESSIO D'OTTAVIO

PHANTOM s.r.l
IMPOSSIBLE IS
OUR TARGET

CONTENTS

TRACCIA

Traccia:

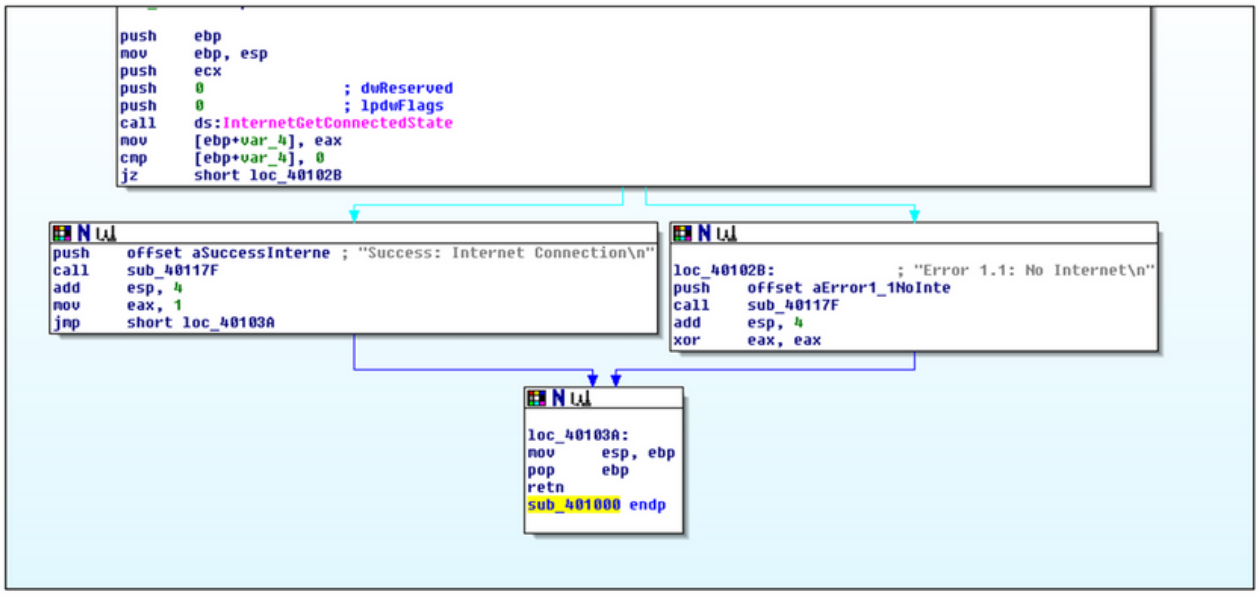
Con riferimento al file **Malware_U3_W2_L5** presente all'interno della cartella «**Esercizio_Pratico_U3_W2_L5** » sul desktop della macchina virtuale dedicata per l'analisi dei malware, rispondere ai seguenti quesiti:

1. Quali **librerie** vengono importate dal file eseguibile?
2. Quali sono le **sezioni** di cui si compone il file eseguibile del malware?

Con riferimento alla figura in slide 3, risponde ai seguenti quesiti:

3. Identificare i **costrutti** noti (creazione dello stack, eventuali cicli, altri costrutti)
4. **Ipotizzare il comportamento della funzionalità implementata**
5. BONUS fare tabella con significato delle singole righe di codice assembly

Figura 1

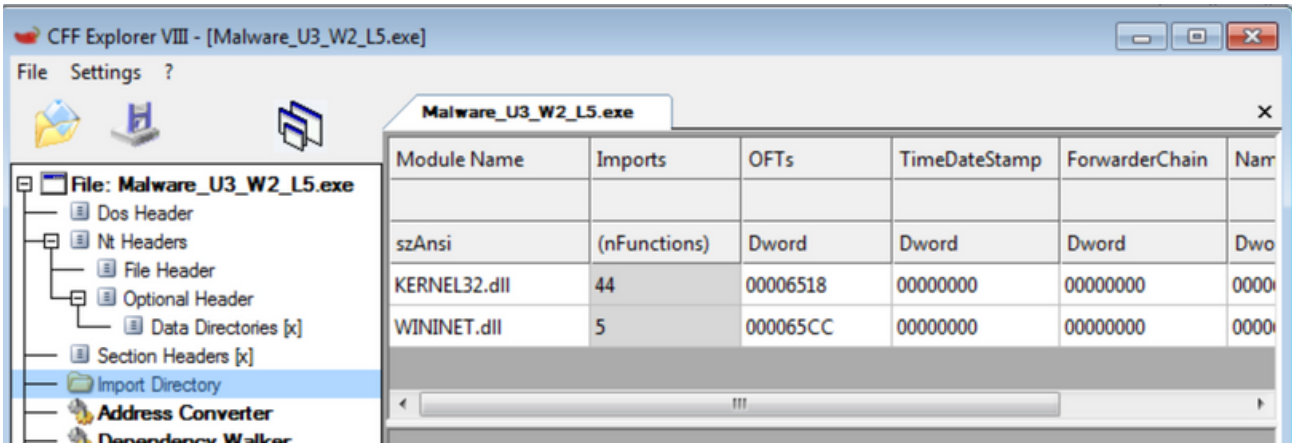


LIBRERIE IMPORTATE

LE LIBRERIE IMPORTATE DAL MALWARE IN QUESTIONE SONO KERNEL32.DLL E WININET.DLL, COMUNEMENTE UTILIZZATE DAI PROGRAMMIMALEVOLI PER SVOLGERE VARIE ATTIVITÀ DANNOSE:

KERNEL32.DLL: QUESTA LIBRERIA FORNISCE FUNZIONALITÀ DI BASE PER LA GESTIONE DELLA MEMORIA, DEI PROCESSI E DEI THREAD ALL'INTERNO DEL SISTEMA OPERATIVO WINDOWS. I MALWARE POSSONO UTILIZZARE LE FUNZIONI ALL'INTERNO DI QUESTA LIBRERIA PER SVOLGERE ATTIVITÀ COME L'AVVIO DI NUOVI PROCESSI, LA GESTIONE DEI FILE E LA MODIFICA DELLE IMPOSTAZIONI DI SISTEMA.

WININET.DLL: QUESTA LIBRERIA FORNISCE FUNZIONALITÀ DI RETE PER APPLICAZIONI WINDOWS, CONSENTENDO LORO DI COMUNICARE SU INTERNET UTILIZZANDO PROTOCOLLI COME HTTP, HTTPS E FTP. I MALWARE POSSONO UTILIZZARE LE FUNZIONI ALL'INTERNO DI QUESTA LIBRERIA PER COMUNICARE CON SERVER REMOTI, SCARICARE E CARICARE FILE, INVIARE DATI RUBATI E PERSINO ESEGUIRE ATTACCHI DI PHISHING.



SEZIONI DEL MALWARE

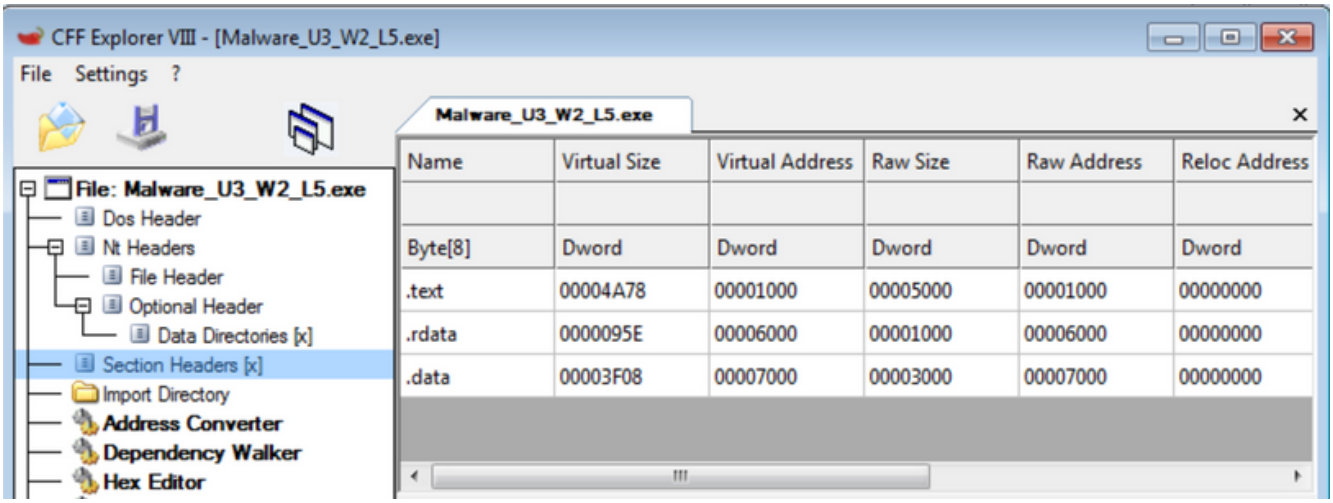
LE SEZIONI DI CUI SI COMPONE QUESTO MALWARE SONO .TEXT, .RDATA E .DATA, ESSE SVOLGONO DIVERSE FUNZIONI NEL PROCESSO DI ESECUZIONE DEL MALWARE. ECCO UNA BREVE SPIEGAZIONE DI CIASCUNA:

.TEXT: QUESTA SEZIONE CONTIENE IL CODICE ESEGUIBILE DEL MALWARE. È DOVE SONO PRESENTI LE ISTRUZIONI CHE VENGONO ESEGUITE QUANDO IL MALWARE VIENE ESEGUITO. QUESTA SEZIONE PUÒ INCLUDERE IL CODICE PRINCIPALE DEL MALWARE, NONCHÉ LE SUBROUTINE E LE FUNZIONI UTILIZZATE PER ESEGUIRE VARIE ATTIVITÀ DANNOSE, COME LA RACCOLTA DI INFORMAZIONI, LA MANIPOLAZIONE DEL SISTEMA E LA COMUNICAZIONE CON SERVER REMOTI.

.RDATA: QUESTA SEZIONE CONTIENE DATI DI SOLA LETTURA CHE SONO INCORPORATI DIRETTAMENTE NEL MALWARE. QUESTI DATI POSSONO INCLUDERE STRINGHE DI TESTO, COSTANTI E ALTRI DATI STATICI UTILIZZATI DAL MALWARE DURANTE L'ESECUZIONE. AD ESEMPIO, POTREBBE INCLUDERE URL DI COMANDO E CONTROLLO, CHIAVI DI CRITTOGRAFIA, INFORMAZIONI DI CONFIGURAZIONE E ALTRO ANCORA.

.DATA: QUESTA SEZIONE CONTIENE DATI MODIFICABILI UTILIZZATI DAL MALWARE DURANTE L'ESECUZIONE. QUESTI DATI POSSONO INCLUDERE VARIABILI, BUFFER, STRUTTURE DATI E ALTRO ANCORA. LA SEZIONE .DATA VIENE SPESSO UTILIZZATA PER MEMORIZZARE INFORMAZIONI DINAMICHE RACCOLTE DURANTE L'ESECUZIONE DEL MALWARE, COME DATI DI SISTEMA, INFORMAZIONI RUBATE DALL'UTENTE O DATI TEMPORANEI UTILIZZATI DURANTE L'ESECUZIONE DELLE ATTIVITÀ DEL MALWARE.

POSSIAMO DUNQUE IPOTIZZARE DA UN'ATTENTA ANALISI CHE SI POSSA TRATTARE DI UN **TROJAN**



COSTRUTTI NOTI

NEL CODICE FORNITO, POSSIAMO IDENTIFICARE DIVERSI COSTRUTTI NOTI:

SETUP DEL FRAME DELLO STACK: LE ISTRUZIONI PUSH EBP E MOV EBP, ESP VENGONO UTILIZZATE PER CONFIGURARE IL FRAME DELLO STACK, CHE È UNA PARTE ESSENZIALE DELLA GESTIONE DELLE CHIAMATE DI FUNZIONE IN ASSEMBLY.

CHIAMATA DI FUNZIONE: LA CHIAMATA CALL DS:INTERNETGETCONNECTEDSTATE INVOCA UNA FUNZIONE CHIAMATA INTERNETGETCONNECTEDSTATE CHE SEMBRA ESSERE UNA PARTE DI UNA LIBRERIA O MODULO ESTERNO.

GESTIONE DEI RISULTATI DELLA FUNZIONE: DOPO LA CHIAMATA DI FUNZIONE, IL VALORE RESTITUITO VIENE CONFRONTATO CON ZERO UTILIZZANDO LE ISTRUZIONI CMP E JZ. QUESTO È UN COMUNE COSTRUTTO DI CONTROLLO DEL FLUSSO PER GESTIRE I RISULTATI DELLE CHIAMATE DI FUNZIONE.

STAMPA DEI MESSAGGI DI OUTPUT: LE ISTRUZIONI PUSH E CALL VENGONO UTILIZZATE PER STAMPARE MESSAGGI DI OUTPUT, SIA PER INDICARE IL SUCCESSO (ASUCCESSINTERNE) CHE PER SEGNALARE ERRORI (AERROR1_1NOINTE).

CLEANUP DEL FRAME DELLO STACK: ALLA FINE DELLA FUNZIONE, IL FRAME DELLO STACK VIENE RIPRISTINATO UTILIZZANDO LE ISTRUZIONI MOV ESP, EBP E POP EBP, ASSICURANDO CHE LO STACK TORNI ALLO STATO INIZIALE PRIMA DELLA CHIAMATA DELLA FUNZIONE.

RITORNO DALLA FUNZIONE: LA FUNZIONE TERMINA CON L'ISTRUZIONE RETN, CHE RIPORTA IL CONTROLLO AL PUNTO IN CUI È STATA CHIAMATA LA FUNZIONE.

IPOTESI SULLA FUNZIONALITÀ

IL CODICE SEMBRA ESSERE UNA PARTE DI UN PROGRAMMA SCRITTO IN LINGUAGGIO ASSEMBLY CHE VERIFICA LO STATO DELLA CONNESSIONE INTERNET E STAMPA UN MESSAGGIO DI SUCCESSO O ERRORE A SECONDA DEL RISULTATO.

IPOTESI SULLA FUNZIONALITÀ DEL CODICE:

VERIFICA DELLO STATO DELLA CONNESSIONE INTERNET: LA FUNZIONE INTERNETGETCONNECTEDSTATE VIENE CHIAMATA PER VERIFICARE SE IL SISTEMA È CONNESSO A INTERNET. IL VALORE RESTITUITO DALLA FUNZIONE VIENE MEMORIZZATO IN [EBP+VAR_4].

GESTIONE DEI RISULTATI: SE IL VALORE RESTITUITO DALLA FUNZIONE È DIVERSO DA ZERO, IL CODICE INTERPRETA CIÒ COME UNA CONNESSIONE INTERNET ATTIVA E STAMPA UN MESSAGGIO DI SUCCESSO. ALTRIMENTI, SE IL VALORE RESTITUITO È ZERO, VIENE STAMPATO UN MESSAGGIO DI ERRORE INDICANTE L'ASSENZA DI CONNESSIONE INTERNET.

STAMPA DEI MESSAGGI DI OUTPUT: I MESSAGGI DI OUTPUT VENGONO STAMPATI UTILIZZANDO UNA FUNZIONE DI STAMPA CHE SEMBRA ESSERE CHIAMATA SUB_40117F.

IN SINTESI, IL CODICE SEMBRA ESSERE PARTE DI UN PROGRAMMA CHE CONTROLLA LO STATO DELLA CONNESSIONE INTERNET E FORNISCE UN FEEDBACK ALL'UTENTE IN BASE A TALE STATO. POTREBBE ESSERE UTILIZZATO PER SCOPI DIAGNOSTICI O PER GARANTIRE CHE UN'APPLICAZIONE FUNZIONI CORRETTAMENTE SOLO QUANDO È DISPONIBILE UNA CONNESSIONE INTERNET ATTIVA.

TRACCIA BONUS

ECCO UNA SPIEGAZIONE DETTAGLIATA DI OGNI RIGA DEL CODICE ASSEMBLY FORNITO:

PUSH EBP: QUESTA ISTRUZIONE METTE IL VALORE CORRENTE DEL REGISTRO DI BASE (EBP) NELLO STACK. QUESTO È UN PASSO COMUNE PER SALVARE LO STATO DEL REGISTRO PRIMA DI MODIFICARLO ALL'INTERNO DI UNA FUNZIONE.

MOV EBP, ESP: QUESTA ISTRUZIONE IMPOSTA IL REGISTRO DI BASE (EBP) AL VALORE CORRENTE DELLO STACK POINTER (ESP). QUESTO È UN PASSO COMUNE PER IMPOSTARE IL FRAME DI BASE PER UNA NUOVA FUNZIONE ALL'INTERNO DEL PROLOGO DELLA FUNZIONE.

PUSH ECX: QUESTA ISTRUZIONE METTE IL VALORE CORRENTE DEL REGISTRO ECX NELLO STACK. POTREBBE ESSERE UTILIZZATO PER SALVARE LO STATO DEL REGISTRO ECX.

PUSH 0 ; DWRESERVED: QUESTA ISTRUZIONE METTE IL VALORE 0 NELLO STACK. QUESTO È IL PRIMO ARGOMENTO PASSATO ALLA FUNZIONE INTERNETGETCONNECTEDSTATE.

PUSH 0 ; 1PDWFLAGS: QUESTA ISTRUZIONE METTE IL VALORE 0 NELLO STACK. QUESTO È IL SECONDO ARGOMENTO PASSATO ALLA FUNZIONE INTERNETGETCONNECTEDSTATE.

CALL DS:INTERNETGETCONNECTEDSTATE: QUESTA ISTRUZIONE CHIAMA LA FUNZIONE INTERNETGETCONNECTEDSTATE. LA CHIAMATA ALLA FUNZIONE AVRÀ COME ARGOMENTI I DUE ZERI PRECEDENTEMENTE INSERITI NELLO STACK.

MOV [EBP+VAR_4], EAX: QUESTA ISTRUZIONE MEMORIZZA IL VALORE RESTITUITO DALLA FUNZIONE INTERNETGETCONNECTEDSTATE NELLA VARIABILE [EBP+VAR_4].

CMP [EBP+VAR_4], 0: QUESTA ISTRUZIONE CONFRONTA IL VALORE MEMORIZZATO NELLA VARIABILE [EBP+VAR_4] CON 0.

JZ SHORT LOC_40102B: QUESTA ISTRUZIONE SALTA A LOC_40102B SE IL RISULTATO DELLA COMPARAZIONE PRECEDENTE È ZERO, IL CHE INDICA CHE NON C'È CONNESSIONE INTERNET.

PUSH OFFSET ASUCCESSINTERNE ; "SUCCESS: INTERNET CONNECTION\N" : QUESTA ISTRUZIONE METTE L'INDIRIZZO DELLA STRINGA "SUCCESS: INTERNET CONNECTION\N" NELLO STACK. QUESTA STRINGA VERRÀ UTILIZZATA PER STAMPARE IL MESSAGGIO DI SUCCESSO.

CALL SUB_40117F: QUESTA ISTRUZIONE CHIAMA LA FUNZIONE SUB_40117F, CHE SI PRESUME STAMPI IL MESSAGGIO DI SUCCESSO.

ADD ESP, 4: QUESTA ISTRUZIONE LIBERA LO STACK DOPO CHE LA FUNZIONE DI STAMPA È STATA CHIAMATA.

MOV EAX, 1: QUESTA ISTRUZIONE IMPOSTA IL REGISTRO EAX A 1, CHE VERRÀ RESTITUITO COME VALORE DI RITORNO DELLA FUNZIONE.

JMP SHORT LOC_40103A: QUESTA ISTRUZIONE SALTA A LOC_40103A, CHE È IL PUNTO DOPO IL BLOCCO DI CODICE CHE GESTISCE IL CASO DI SUCCESSO DELLA CONNESSIONE INTERNET.

TRACCIA BONUS

LOC_40102B ; "ERROR 1.1: NO INTERNET\N: ETICHETTA PER INDICARE IL PUNTO NEL CODICE DOVE SI GESTISCE IL CASO IN CUI NON C'È CONNESSIONE INTERNET.

PUSH OFFSET AERROR1_1NOINTE: QUESTA ISTRUZIONE METTE L'INDIRIZZO DELLA STRINGA "ERROR 1.1: NO INTERNET\N" NELLO STACK. QUESTA STRINGA VERRÀ UTILIZZATA PER STAMPARE IL MESSAGGIO DI ERRORE RELATIVO ALLA MANCANZA DI CONNESSIONE INTERNET.

CALL SUB_40117F: QUESTA ISTRUZIONE CHIAMA LA FUNZIONE SUB_40117F, CHE SI PRESUME STAMPI IL MESSAGGIO DI ERRORE.

ADD ESP, 4: QUESTA ISTRUZIONE LIBERA LO STACK DOPO CHE LA FUNZIONE DI STAMPA È STATA CHIAMATA.

XOR EAX, EAX: QUESTA ISTRUZIONE IMPOSTA IL REGISTRO EAX A 0. QUESTO VALORE VERRÀ RESTITUITO COME VALORE DI RITORNO DELLA FUNZIONE IN CASO DI ERRORE.

LOC_40103A: QUESTA È UN'ETICHETTA CHE INDICA IL PUNTO NEL CODICE IN CUI VIENE ESEGUITA L'ISTRUZIONE DOPO CHE IL BLOCCO DI CODICE RELATIVO AL CONTROLLO DELLA CONNESSIONE INTERNET È STATO ESEGUITO, INDIPENDENTEMENTE DAL RISULTATO.

MOV ESP, EBP: QUESTA ISTRUZIONE RIPRISTINA IL VALORE ORIGINALE DELLO STACK POINTER (ESP) PER IL FRAME DI CHIAMATA CORRENTE, CIOÈ PRIMA CHE IL REGISTRO DI BASE (EBP) FOSSE IMPOSTATO.

POP EBP: QUESTA ISTRUZIONE RIPRISTINA IL REGISTRO DI BASE (EBP) AL SUO VALORE ORIGINALE PRIMA DELLA CHIAMATA DELLA FUNZIONE.

RETN: QUESTA ISTRUZIONE INDICA IL RITORNO DALLA FUNZIONE, RESTITUENDO IL CONTROLLO AL CHIAMANTE E UTILIZZANDO IL VALORE ATTUALMENTE NELLO STACK COME VALORE DI RITORNO.

SUB_401000 ENDP: L'ETICHETTA ENDP INDICA LA FINE DELLA PROCEDURA. QUINDI, QUESTA RIGA SEGNA LA FINE DEL BLOCCO DI CODICE DELLA FUNZIONE SUB_401000. LA FUNZIONE TERMINA QUI E IL CONTROLLO VIENE RESTITUITO AL PUNTO IN CUI È STATA CHIAMATA.

PREPARED BY
PHANTOM SRL

29/03/2024

GRAZIE!

D'OTTAVIO ALESSIO