

PREAPRED BY :
PHANTOM SRL

Pignatello Giuseppe
D'Ottavio Alessio
Iannone Luca



PHANTOM s.r.l.
IMPOSSIBLE IS
OUR TARGET

2024

REPORT

INDICE

Page 3: Traccia Esercizio

Page 4: Cosa sono le chiavi di registro?

Page 5: Risoluzione

Page 6: Traccia Bonus



ASSEMBLY

CHIAVI DI REGISTRO

Il Registro di Windows è un componente cruciale del sistema operativo che memorizza i dati di configurazione delle applicazioni, dei dispositivi hardware e delle impostazioni di sistema.

Contiene informazioni sui programmi installati, sulle preferenze dell'utente, sui driver dei dispositivi e su altri dati critici.

A volte, gli utenti possono avere bisogno di accedere o modificare le chiavi di registro per risolvere problemi, risolvere errori o migliorare le prestazioni.

Come trovare le chiavi di registro?

Il Registro di Windows è un database gerarchico organizzato in chiavi, sottochiavi e valori. Ogni programma o applicazione installata sul computer ha una propria serie di chiavi di registro che contengono varie impostazioni e configurazioni. Ecco come trovare le chiavi di registro per un programma specifico:

1. Premere contemporaneamente il tasto Windows e R per aprire la finestra di dialogo **Esegui**.

2. Digitare "**regedit**" nella casella Esegui e fare clic su **OK**.

Il registro di Windows si divide in **5** macrocategorie:

- 1)HKEY_LOCAL_MACHINE
- 2)HKEY_CURRENT_USER
- 3)HKEY_CLASSES_ROOT
- 4)HKEY_CURRENT_CONFIG
- 5)HKEY_USERS



RISOLUZIONE

Persistenza:

Nel codice fornito, la **persistenza** avviene tramite l'uso della funzione **RegSetValueExW**, che imposta un valore per una chiave nel registro di sistema. La chiave del registro interessata è

"Software\Microsoft\Windows\CurrentVersion\Run", che è comunemente utilizzata per avviare programmi all'avvio del sistema operativo.

Questo frammento di codice apre la chiave del registro

"Software\Microsoft\Windows\CurrentVersion\Run" con **RegOpenKeyExW**, e successivamente chiama **RegSetValueExW** per impostare un valore all'interno di quella chiave. Questo processo di scrittura di un valore all'interno della chiave "Run" del registro di sistema è un'indicazione comune di persistenza, poiché i programmi che vengono aggiunti a questa chiave verranno eseguiti all'avvio del sistema operativo.

Client Software:

Dal codice assembly, possiamo affermare che il "client software" è progettato per stabilire una connessione a Internet utilizzando le API di Windows, in particolare le funzioni **InternetOpenA** e **InternetOpenUrlA**.

La stringa **"Internet Explorer 8.0"** passata come parametro **IpszAgent** nella chiamata **InternetOpenA** suggerisce che il software potrebbe tentare di impersonare l'agente utente di Internet Explorer. Questo viene fatto per ottenere risposte dai server web che possono variare in base all'agente utente.

Il ciclo infinito presente nel codice indica che il software potrebbe essere progettato per eseguire un'azione ripetuta, come ad esempio il polling di un'URL specifico o il download periodico di risorse da Internet.

URL:

L'URL utilizzato nelle chiamate di funzione è **"http://www.malware12com/"**.

TRACCIA BONUS



• BONUS: qual è il significato e il funzionamento del comando assembly "lea"?

LEA è l'abbreviazione di "Load Effective Address" (Carica indirizzo effettivo). Carica l'indirizzo del riferimento alla posizione dall'operando di origine all'operando di destinazione ed è possibile utilizzarlo per effettuare molteplici azioni:

Somma di registri: `lea edx, [eax + ebx]` - Calcola l'indirizzo effettivo della somma dei contenuti dei registri `eax` e `ebx`.

Moltiplicazione per costante: `lea esi, [ebx * 2]` - Calcola l'indirizzo effettivo moltiplicando il contenuto del registro `ebx` per 2.

Somma con offset: `lea edi, [eax + ebx + 10]` - Calcola l'indirizzo effettivo della somma dei contenuti dei registri `eax` e `ebx`, a cui viene aggiunto un offset di 10.

Somma con moltiplicazione: `lea esi, [eax + ebx * 4]` - Calcola l'indirizzo effettivo della somma del contenuto del registro `eax` e del contenuto del registro `ebx` moltiplicato per 4.

Somma di variabili: `lea edx, [var1 + var2]` - Calcola l'indirizzo effettivo della somma delle variabili `var1` e `var2`.

Accesso agli elementi di un array: `lea ecx, [eax + ebx * 4]` - Calcola l'indirizzo effettivo dell'elemento dell'array indicizzato da `ebx` (utilizzato spesso in linguaggi ad alto livello come C/C++ per l'accesso agli array).

Calcolo di indirizzi in strutture di dati complesse: `lea edx, [eax + 2*ebx + struct_ptr]` - Calcola l'indirizzo effettivo di un campo all'interno di una struttura di dati, dove `struct_ptr` punta alla struttura stessa.

In generale, l'istruzione `lea` è una delle istruzioni più potenti ed utili in assembly per calcolare indirizzi in modo efficiente, consentendo di eseguire operazioni aritmetiche complesse e combinazioni su indirizzi senza accedere direttamente alla memoria.

Nel codice assembly fornito, l'istruzione `lea` viene utilizzata per calcolare un indirizzo effettivo e caricarlo in un registro. Questa istruzione non accede direttamente alla memoria, ma calcola l'indirizzo senza effettuare l'accesso effettivo.

9T@5Answer



lea assembly instruction

PREAPRED BY :
PHANTOM SRL

Pignatello Giuseppe
D'Ottavio Alessio
Iannone Luca



PHANTOM s.r.l.
IMPOSSIBLE IS
OUR TARGET

2024

GRAZIE