Malware analysis

Scaricare il malware presente in questo link, effettuare un'analisi completa e creare un report:
https://github.com/Da2dalus/The-MALWARERepo/blob/master/rogues/AdwereCleaner.exe

Anyrun

Studiare questi link di anyrun e spiegare queste minacce in un piccolo report:
https://app.any.run/tasks/371957e1-d9604b8a-8c68241ff918517d/
https://app.any.run/tasks/f1f20828222246fb-a88609f77581e67b/

Come output vorrei la spiegazione in italiano per un eventuale cliente / manager (che è poco preparato sulla materia) di questi malware (o presunti tali). Indicare le vostre scelte di remediation (mettere in quarantena, eliminare, blacklist, falso positivo, falso negativo, vero positivo, vero negativo, chiedo al vendor, ecc.) motivandole.

Lab - Navigating the Linux Filesystem and Permission Settings

In this lab, you will familiarize yourself with Linux filesystems

Lab - Extract an Executable from a PCAP

Looking at logs is very important, but it is also important to understand how network transactions happen at the packet level. In this lab, you will complete the following objective:

- Analyze the traffic in a previously captured pcap file and extract an executable file from the traffic.

Bonus 1

Altro link di Anyrun da studiare e spiegare in un report.

Bonus 2

Lab - Interpret HTTP and DNS Data to Isolate Threat Actor

In this lab, you will complete the following objective:

- Investigate SQL injection and DNS exfiltration exploits using Security Onion tools.

Bonus 3

Lab - Isolate Compromised Host Using 5-Tuple

In this lab, you will complete the following objective:

- Use Security Onion tools to investigate an exploit.

Bonus 4

Il malware Mydoom, apparso per la prima volta nel 2004, è uno dei worm più distruttivi della storia informatica. Si è diffuso principalmente tramite e-mail, infettando i computer Windows e lasciando aperte le porte di rete per future intrusioni. Il suo rapido spread ha causato gravi rallentamenti su reti aziendali e Internet globalmente.

1. Analisi Forense: Attraverso l'analisi del codice sorgente, potete imparare come funziona un malware dal punto di vista tecnico. Questo include l´analisi delle funzioni di propagazione, le tecniche di evasione dei sistemi di sicurezza, e la comprensione di come il malware gestisce la comunicazione con i server di comando e controllo.

2. Scenario di Intelligence: Supponiamo che la nostra intelligence abbia scoperto una nuova variante di Mydoom che sta emergendo. Dovete valutare il codice per identificare possibili modifiche o aggiornamenti rispetto alla versione originale. Questo esercizio mira a preparavi a rispondere rapidamente a nuove minacce, sviluppando capacità di analisi critica e di adattamento a scenari di sicurezza in evoluzione.

Bonus 5

In questo esercizio, affronteremo una richiesta realistica di un cliente che ha identificato una vulnerabilità di tipo buffer overflow nella sua applicazione principale. L'obiettivo è redigere un report dimostrativo che illustri come un attaccante potrebbe sfruttare questa vulnerabilità.

Obiettivi dell'Esercizio:

- Comprensione della Vulnerabilità: Iniziate analizzando il concetto di buffer overflow, comprendendo come questa vulnerabilità possa essere sfruttata per eseguire codice arbitrario sul sistema colpito.
- Preparazione dell'Ambiente di Test: Utilizzate un ambiente controllato per replicare un'applicazione che simuli le condizioni di vulnerabilità simile a quella del cliente.

Sviluppo di un exploit:

- Analisi: Determinare la dimensione del buffer e il punto preciso dove il buffer overflow causa il crash dell'applicazione.
- Creazione del Payload: Usate strumenti come Metasploit o script in Python per creare un payload che sfrutti la vulnerabilità.
- Test dell'Exploit: Eseguite l'exploit nell'ambiente di test per confermare l'esecuzione di codice arbitrario.
- Proposte di Mitigazione e Raccomandazioni: Dopo aver dimostrato l'exploit, proponete soluzioni per mitigare la vulnerabilità. Questo include l'aggiornamento del codice, l'applicazione di patch di sicurezza e l'adozione di buone pratiche di programmazione sicura.

Il vostro report finale dovrà includere:

Una introduzione alla vulnerabilità e al contesto dell'applicazione. I dettagli tecnici dell'analisi del buffer overflow e dello sviluppo dell'exploit. Una dimostrazione dell'exploit, supportata da screenshot o video. Raccomandazioni per la mitigazione, con spiegazioni dettagliate delle soluzioni proposte.