Bonus 1: Interpretare Dati HTTP e DNS per Isolare l'Attore della Minaccia

Obiettivi

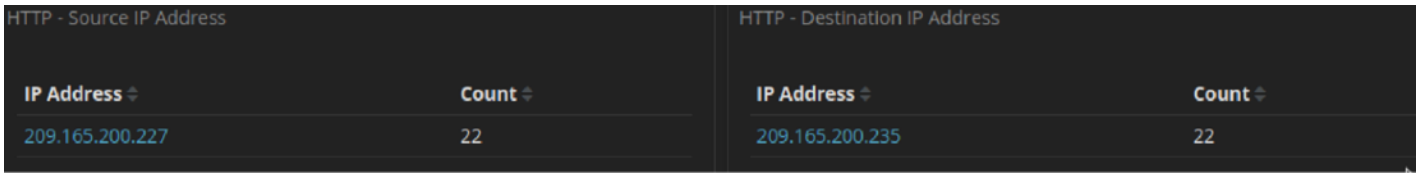
In questo laboratorio, esaminerai i log di uno sfruttamento di vulnerabilità documentate HTTP e DNS.

- Parte 1: Investigare un Attacco di SQL Injection
- Parte 2: Investigare l'Esfiltrazione di Dati DNS

Domande

1.Qual è l'indirizzo IP sorgente?
2Qual è l'indirizzo IP destinazione?

IP sorgente 209.165.200.227
IP destinazione 209.168.200.235



HTTP - Source IP Address

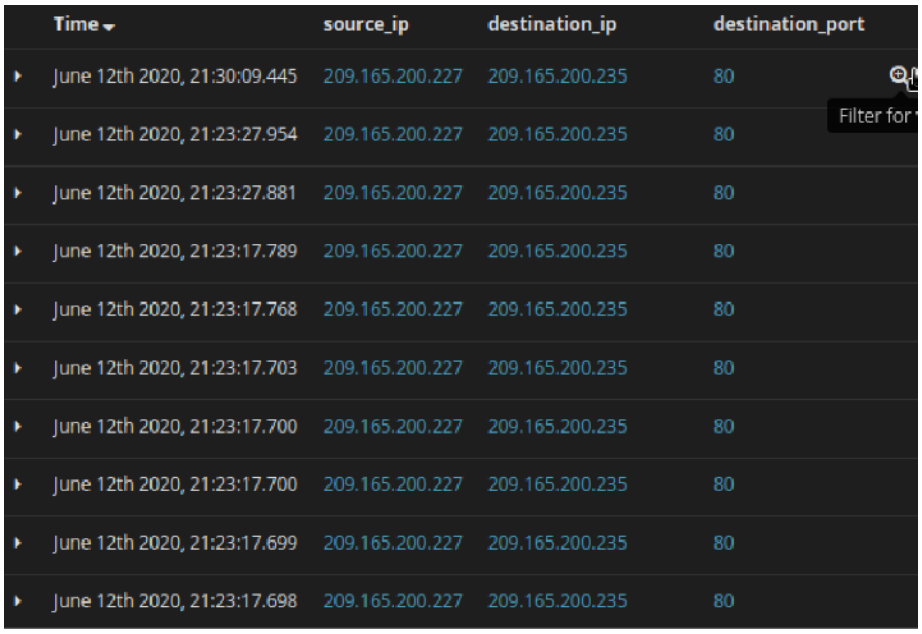
IP Address	Count
209.165.200.227	22

HTTP - Destination IP Address

IP Address	Count
209.165.200.235	22

3.Qual è il numero di porta destinazione?

La porta di destinazione è la porta 80



Time	source_ip	destination_ip	destination_port
June 12th 2020, 21:30:09.445	209.165.200.227	209.165.200.235	80
June 12th 2020, 21:23:27.954	209.165.200.227	209.165.200.235	80
June 12th 2020, 21:23:27.881	209.165.200.227	209.165.200.235	80
June 12th 2020, 21:23:17.789	209.165.200.227	209.165.200.235	80
June 12th 2020, 21:23:17.768	209.165.200.227	209.165.200.235	80
June 12th 2020, 21:23:17.703	209.165.200.227	209.165.200.235	80
June 12th 2020, 21:23:17.700	209.165.200.227	209.165.200.235	80
June 12th 2020, 21:23:17.700	209.165.200.227	209.165.200.235	80
June 12th 2020, 21:23:17.699	209.165.200.227	209.165.200.235	80
June 12th 2020, 21:23:17.698	209.165.200.227	209.165.200.235	80

Filter for

4.Qual è il timestamp del primo risultato?

Timestamp June 12th 2020, 21:30:09.445

5.Qual è il tipo di evento?

L'evento è di tipo DOC

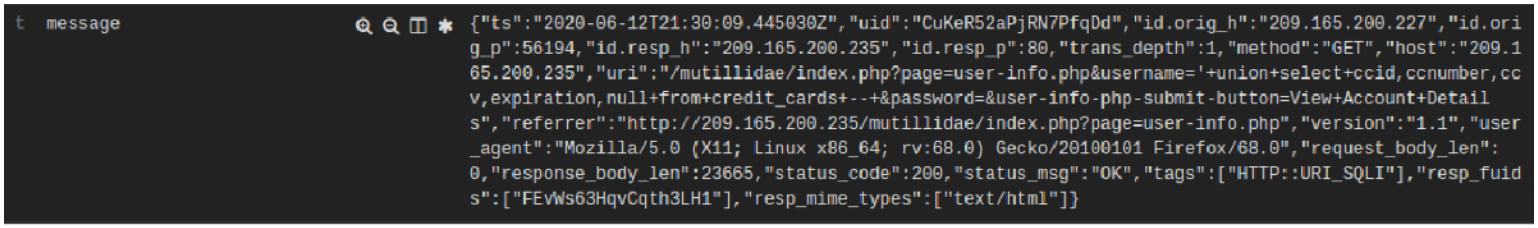
6.Cosa è incluso nel campo message?

Il campo message contiene un record di log dettagliato in formato JSON che cattura una singola transazione HTTP. Questi sono i dettagli della richiesta HTTP GET inviata dal client (id.orig_h: 209.165.200.227) al server (id.resp_h: 209.165.200.235

Il campo più critico per l'analisi è l'**URI** (Uniform Resource Identifier), che rivela la stringa di query inviata al server.

L'URI non è una richiesta di navigazione normale. La sua struttura evidenzia inequivocabilmente un tentativo di **SQL Injection (SQLi)**, una vulnerabilità di sicurezza che permette all'attaccante di interferire con le query che un'applicazione web esegue sul suo database.

1. **Obiettivo della Pagina:** Il percorso punta a una pagina di gestione utente: `/multillidae/index.php?page=user-info.php`.
2. **Payload (Carico Utile):** L'attacco è contenuto nel parametro `username`: 'username'='+union+select+ccid,ccnumber,ccv,expiration,null+from



7.Qual è il significato di queste informazioni?

Il significato di queste informazioni è che è stato rilevato un **tentativo di attacco di SQL Injection (SQLi)** in tempo reale.

1. **Conferma dell'Attacco:** Il tag **`"URI_SQLI"`** (Uniform Resource Identifier - SQL Injection) e la presenza del comando **`UNION SELECT`** nella stringa URI confermano l'attacco.
2. **Obiettivo Dati:** La stringa iniettata (+union+select+ccid,ccnumber,ccv,expiration,null+from+credit_cards--) rivela che l'obiettivo specifico dell'attaccante è l'**esfiltrazione di dati sensibili di carte di credito** dalla tabella di database denominata credit_cards.
3. **Stato del Server:** Il codice di stato **`200`** indica che il server web ha elaborato la richiesta con successo. Questo solleva una preoccupazione critica: **se l'attacco è andato a buon fine, i dati delle carte di credito potrebbero essere stati restituiti** nel corpo della risposta al client.

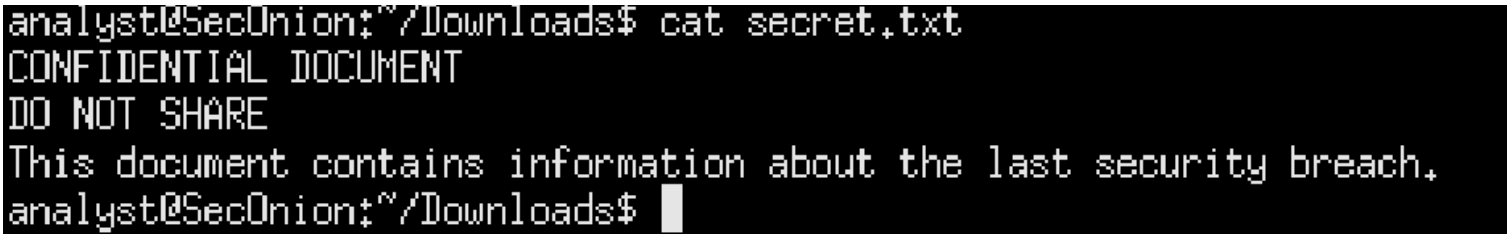
8.Cosa vedi più avanti nella trascrizione riguardo ai nomi utente?

Si puo vedere che l'attaccante è riuscito ad estrarre dei Username, Password e delle Segnature

```
DST: <b>Username=</b>4444111122223333<br>
DST:
DST: 17
DST: <b>Password=</b>745<br>
DST:
DST: 22
DST: <b>Signature=</b>2012-03-01<br><p>
DST:
DST: 24
DST: <b>Username=</b>7746536337776330<br>
DST:
DST: 17
DST: <b>Password=</b>722<br>
DST:
DST: 22
DST: <b>Signature=</b>2015-04-01<br><p>
DST:
DST: 24
DST: <b>Username=</b>8242325748474749<br>
DST:
DST: 17
DST: <b>Password=</b>461<br>
DST:
DST: 22
DST: <b>Signature=</b>2016-03-01<br><p>
DST:
```

9.I sottodomoni delle quei DNS erano sottodomini? Se no, qual è il testo?

Il testo come si puo vedere è:



10.Cosa implica questo risultato riguardo a queste particolari richieste DNS? Qual è il significato più ampio?

Le richieste DNS contenenti lunghe stringhe esadecimali che abbiamo analizzato **non erano normali richieste di risoluzione di dominio**. Erano dati codificati che provano un attacco di **DNS Tunneling**.

1. Tecnica d'Attacco

- **DNS Tunneling:** Un metodo stealth (nascosto) per far uscire dati dalla rete, inserendoli nel campo del sottodominio delle richieste DNS (porta 53).

2. Risultato e Prova della Violazione

- **Esfiltrazione Confermato:** La decodifica delle stringhe esadecimali ha recuperato un **documento confidenziale** rubato.
- Messaggio Esfiltrato: Il messaggio segreto era: "CONFIDENTIAL DOCUMENT DO NOT SHARE This document contains information about the last security breach."

3. Implicazioni Critiche

- **Massima Gravità:** L'attacco è riuscito a **eludere i normali firewall** e i sistemi di prevenzione della perdita di dati (DLP).

- **Azione Necessaria:** Questo è un evento di sicurezza critico che richiede l'**immediato isolamento dell'host infetto** e l'implementazione di sistemi di sicurezza più robusti (come un DNS Firewall o sistemi NDR) per ispezionare il traffico DNS in uscita.

11.Cosa potrebbe aver creato queste query DNS codificate e perché è stato scelto il DNS come mezzo per esfiltrare dati?

Le query DNS codificate sono state generate da un **client di DNS Tunneling**, ovvero un'applicazione malevola installata sull'host interno compromesso.

- Questo strumento ha suddiviso il documento riservato in piccoli blocchi, li ha **codificati in esadecimale** e li ha inviati all'esterno, utilizzando il campo del sottodominio come contenitore per i dati rubati.

Il DNS è stato scelto per questo attacco perché offre il massimo grado di **stealth** per due ragioni:

- **Bypass del Firewall:** Il traffico DNS è un protocollo **vitale** per qualsiasi rete e per default è quasi sempre **consentito** dai firewall in uscita. Questo permette ai dati codificati di passare inosservati.

- **Mancanza di Ispezione:** La maggior parte dei sistemi di sicurezza non ispeziona il payload del traffico DNS in modo rigoroso. L'attaccante sfrutta questa **fiducia** per utilizzare il DNS come un canale di comunicazione nascosto per l'esfiltrazione di dati sensibili e per il Comando e Controllo (C2).