LABORATORIO DI INTERPRETAZIONE DATI HTTP E DNS PER ISOLARE UN ATTACCANTE

1. Introduzione

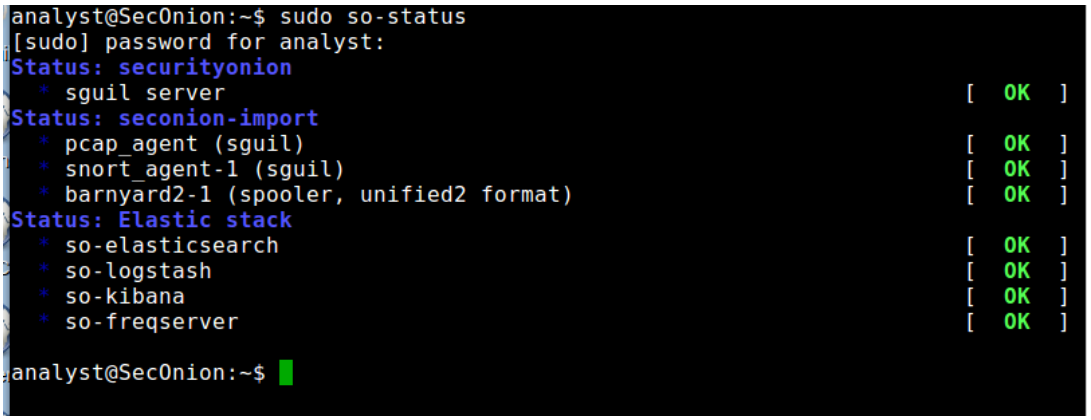
Il presente laboratorio ha lo scopo di esaminare come agisce un exploit SQL Injection in cui è stato eseguito un accesso non autorizzato ad informazioni sensibili archiviate su un Web Server.
Un attacco SQL Injection è una vulnerabilità di sicurezza delle applicazioni web che permette ad un attaccante di inserire comandi SQL malevoli in una query SQL eseguita dal database. Lo scopo di questo tipo di attacco è quello di manipolare le query alterando il comportamento previsto ed ottenendo l'accesso non autorizzato ai dati o alle funzionalità di una applicazione.
A tale scopo verrà utilizzato Kibana, uno strumento utile alla visualizzazione di dati, la sua funzione principale è quella di esplorare, analizzare e rappresentare graficamente grandi quantità di dati raccolti ed indicizzati in Elasticsearch, un motore di ricerca distribuito.

2. Modifica dell'intervallo di tempo

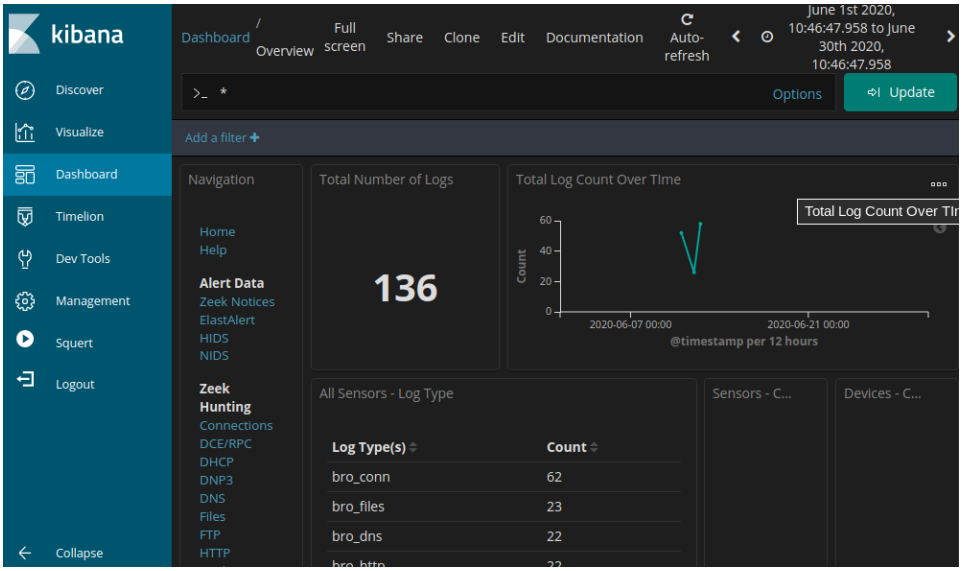
La Virtual Machine che verrà utilizzata in questo laboratorio è la Security Onion, macchina virtuale creata allo scopo di permettere la scoperta e l'utilizzo dei principali strumenti utili ad un operatore appartenente al SOC per l'analisi delle minacce, malware ed exploit in un sistema informatico.

Il primo passaggio importante sarà quello di controllare lo stato dei servizi in maniera tale da determinare se la macchina virtuale è operativa per l'analisi, per fare ciò si utilizza il comando nel terminale:

sudo so-status



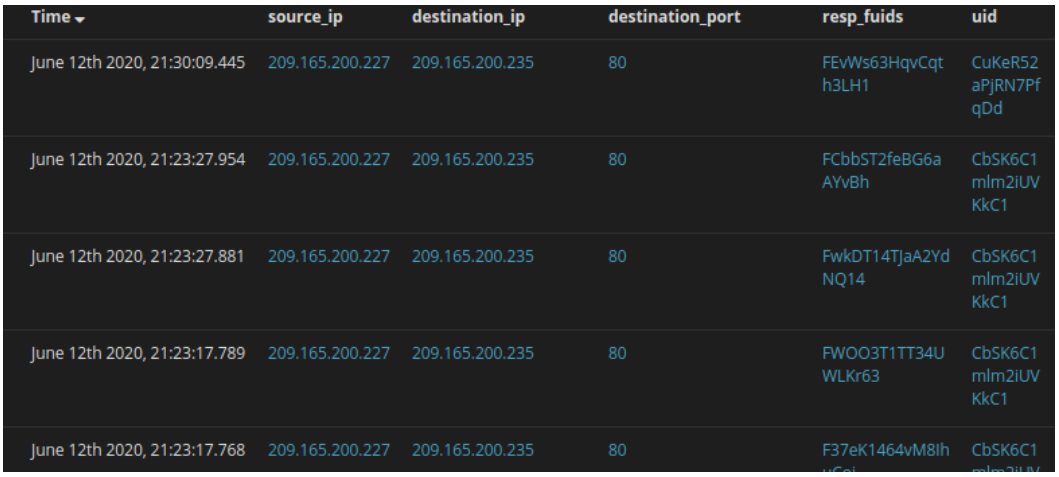
Se l'output non restituisce messaggi di errore, sarà possibile procedere con l'analisi eseguita con Kibana.
Si andrà ad utilizzare il comando rapido presente sul Desktop di Security Onion, le credenziali utilizzate sono le stesse della VM. Kibana offre la possibilità di personalizzare la Dashboard in base ai monitoraggi da effettuare.
L'attacco che sarà analizzato ha avuto luogo a Giugno 2020, la prima operazione da effettuare all'interno di Kibana sarà impostare le date interessate da questo attacco, sarà impostato il mese di Giugno 2020 per intero.
Tornando sulla Dashboard, sarà possibile visualizzare un totale di 136 interazioni di log nell'intervallo di tempo selezionato.



3. Filtraggio del traffico HTTP

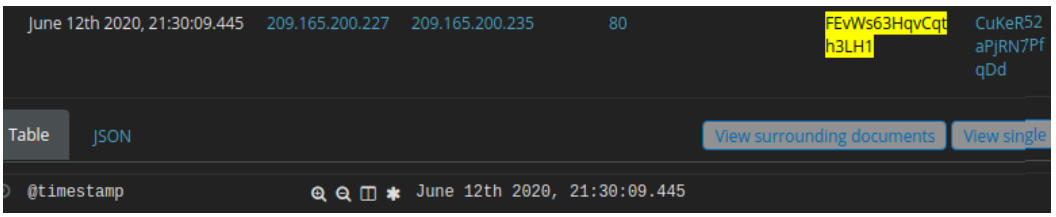
Essendo un attacco rivolto ad un Web Server, la cosa più utile da fare sarà sicuramente una analisi basata sul filtro HTTP che ci rivelerà i log di traffico che utilizzano questo protocollo, quella riservata alle comunicazioni web.
Scorrendo la Dashboard filtrata potremo ricavare preziose informazioni quali:

- Indirizzo IP di origine 209.165.200.277;
- Indirizzo IP di destinazione 209.165.200.235;
- Porta di destinazione 80.



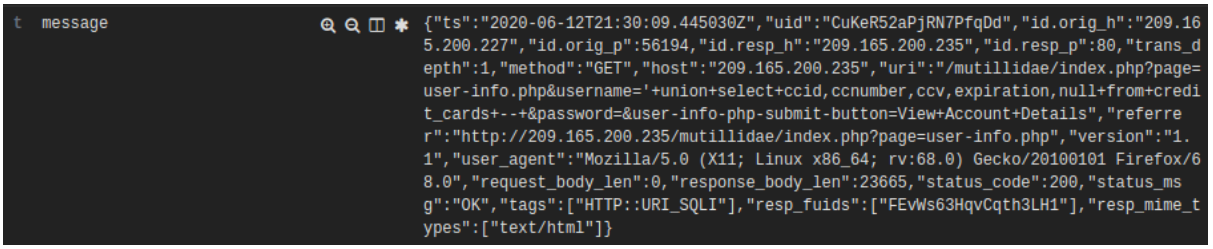
Time	source_ip	destination_ip	destination_port	resp_fuids	uid
June 12th 2020, 21:30:09.445	209.165.200.227	209.165.200.235	80	FEvWs63HqvCqth3LH1	CuKeR52aPjRN7PfqDd
June 12th 2020, 21:23:27.954	209.165.200.227	209.165.200.235	80	FCbbST2feBG6aAYvBh	CbSK6C1mlm2iUVKkC1
June 12th 2020, 21:23:27.881	209.165.200.227	209.165.200.235	80	FwkDT14TjaA2YdNQ14	CbSK6C1mlm2iUVKkC1
June 12th 2020, 21:23:17.789	209.165.200.227	209.165.200.235	80	FWOO3T1TT34UWLKr63	CbSK6C1mlm2iUVKkC1
June 12th 2020, 21:23:17.768	209.165.200.227	209.165.200.235	80	F37eK1464vM8Ih	CbSK6C1

Per effettuare una analisi più approfondita si andrà a guardare i registri di log HTTP, si espanderanno i dettagli riguardanti il primo risultato.
Una informazione importante sarà sicuramente il Timestamp, ovvero data e ora in cui è avvenuta quella particolare interazione, in questo caso risulta il giorno 12 Giugno 2020 alle ore 21:30.



Sicuramente è importante conoscere il tipo di evento, che rivela il nome di *bro_http*.

All’interno del messaggio sarà possibile anche notare la query SQL inserita allo scopo di attingere ai dati senza autorizzazione, nel campo *uri*. La query, in sostanza, chiede di attingere a dati come Username, CCID, numero Conto Corrente, CCV, scadenza e password.

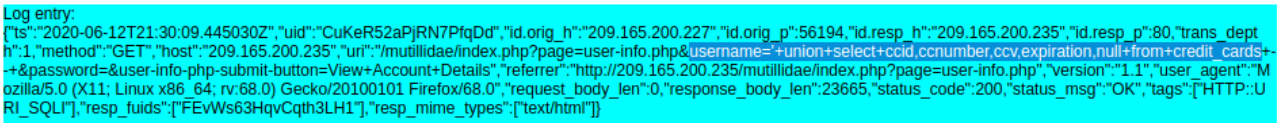


Ciò che si può dedurre da questi dati sarà che questo attacco è improntato sulla ricerca di informazioni sensibili quali le credenziali ed il numero di una carta di credito.

4. **Visualizzazione dei risultati**

Alcune delle informazioni per le voci di registro sono collegate tramite collegamento hypertext ad altri strumenti utili per la loro analisi.
Alla voce *_id* sarà possibile avere accesso ad una visualizzazione più dettagliata dell’evento.

Aprendolo, verrà visualizzata una nuova scheda browser con le informazioni analizzate dall’interfaccia *capME!*, la quale serve a visionare una trascrizione *.pcap*. La parte scritta in blu contiene le richieste HTTP inviate dalla sorgente ed il testo in rosso riporta le risposte del Web Server di destinazione.
Nella sezione dedicata al Log Entry sarà visibile la query SQL incriminata che potrebbe indicare un attacco al browser tramite SQL Injection, facilmente riconoscibile dai termini *union* e *select*.



Ciò accade nel caso in cui le caselle di input su una pagina web non sono state sanitizzate a dovere da impedire l’inserimento di codice illegale.

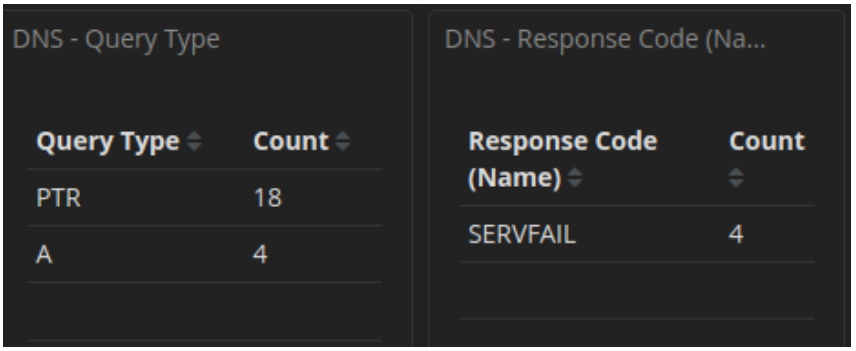
Con una ricerca approfondita sarà possibile individuare facilmente tutti i dati richiesti dalla query presenti su questo database. Il filtro utilizzato sarà *username*.

Da questa lista possiamo facilmente notare che saranno presenti dati come:

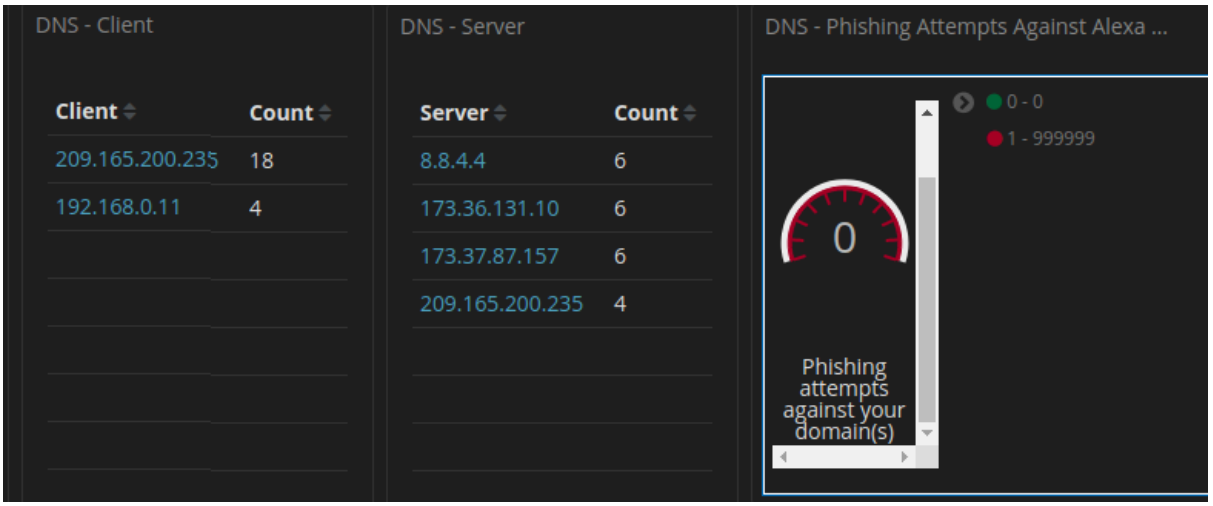
- Nome utente;
- Password;
- Firma.

5. **Analisi del traffico DNS**

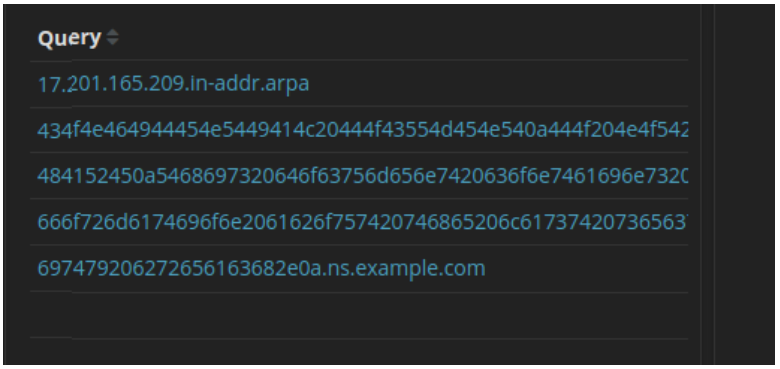
Un amministratore di rete ha notato delle query DNS eccezionalmente lunghe con sottodomini insoliti. Per indagare sull'anomalia bisognerà filtrare il traffico DNS, in maniera analoga a come è stato fatto per il traffico HTTP.
Scorrendo verso il basso sarà possibile visualizzare le principali query DNS.



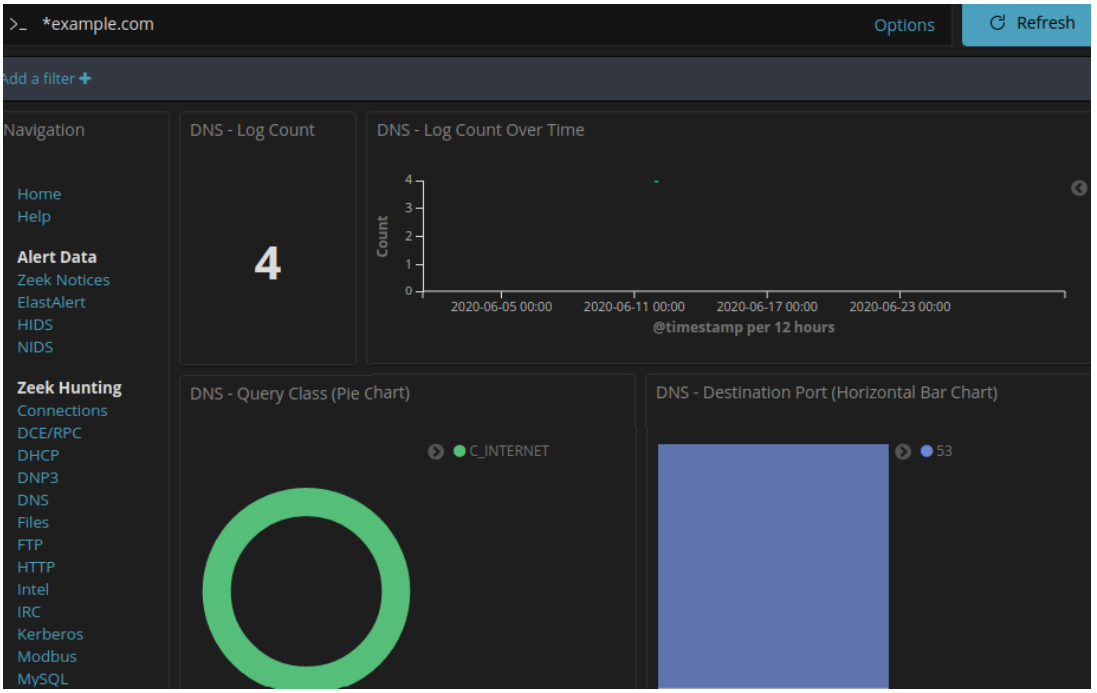
Scorrendo ancora, sarà possibile vedere un elenco dei principali client DNS Phishing (pharming, spoofing o poisoning).



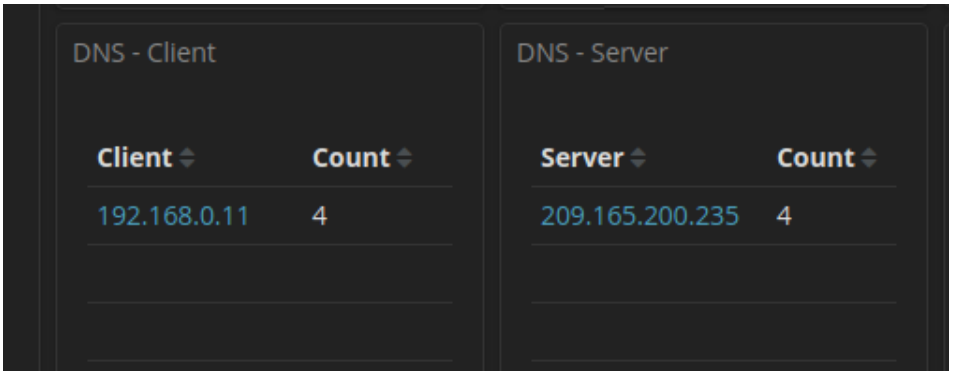
Scendendo ulteriormente in basso si può notare l'elenco delle query principali per nome di dominio, tra quelli insolitamente lunghi potremo notare un dominio *example.com* che potrebbe risultare sospetto e necessitare di approfondimenti.



A questo scopo si effettuerà una ulteriore ricerca specifica sul dominio incriminato. Si andrà ad inserire il dominio nella barra di ricerca in maniera tale da rilevare solo le interazioni attinenti, il risultato sarà quello di visualizzare 4 interazioni.



Successivamente si osserveranno gli indirizzi IP su cui queste interazioni hanno avuto luogo.



6. Analisi dei dati ricavati

Tornando all'elenco delle query rilevate, si andrà ad esportare i dati per una analisi più dettagliata delle query sospette. Il file ricavato sarà il seguente.

```
Query,Count
"434f4e464944454e5449414c20444f43554d454e540a444f204e4f542053.ns.example.com",1
"484152450a5468697320646f63756d656e7420636f6e7461696e7320696e.ns.example.com",1
"666f726d6174696f6e2061626f757420746865206c617374207365637572.ns.example.com",1
"697479206272656163682e0a.ns.example.com",1
```

Si modificherà tale file eliminando il testo che circonda la parte esadecimale dei sottodomini, lasciando solo i caratteri esadecimali, salvandolo poi conservando il nome originale. Dopodichè verrà aperto il terminale per eseguire il comando *xxd*, utile alla decodifica del testo, salvarlo con un altro nome (in questo caso *secret.txt*) ed il comando *cat* per eseguire il contenuto.

```
analyst@SecOnion:~/Downloads$ xxd -r -p "DNS - Queries.csv" > secret.txt
analyst@SecOnion:~/Downloads$ cat secret.txt
CONFIDENTIAL DOCUMENT
DO NOT SHARE
This document contains information about the last security breach.
analyst@SecOnion:~/Downloads$
```

L'output che otterremo sarà un messaggio che comunica di non condividere tali informazioni in quanto riservate e riguardanti l'ultimo attacco.
In poche parole le query DNS potrebbero essere utilizzate per nascondere l'invio di file ed aggirare la sicurezza della rete.

C’è la possibilità che questo malware stia creando queste richieste scorrendo i documenti presenti sull’host e codificando il contenuto in esadecimale, creando query DNS che utilizzano le stringhe esadecimali come sottodomini. Le richieste DNS sono spesso inviate da una rete ad Internet, quindi potrebbero non essere monitorate con assiduità.

7. Conclusioni

Il laboratorio ha illustrato come un attacco SQL Injection possa compromettere la sicurezza di un sistema, consentendo l'accesso non autorizzato a informazioni sensibili archiviate su un Web Server. Utilizzando strumenti come Kibana e la piattaforma Security Onion, abbiamo esplorato il processo di monitoraggio e analisi di traffico sospetto, con particolare attenzione al filtraggio del traffico HTTP e DNS. Attraverso l'uso di Kibana, è stato possibile esaminare i log relativi all'attacco, individuando query SQL malevoli che miravano a estrarre dati sensibili come credenziali, numeri di carte di credito e altre informazioni finanziarie. Inoltre, il filtraggio delle query DNS ha rivelato ulteriori dettagli riguardanti tentativi di esfiltrazione di dati attraverso query DNS manipolate, utilizzate per aggirare i meccanismi di sicurezza della rete.
L'attività ha evidenziato l'importanza della corretta sanitizzazione degli input nelle applicazioni web per prevenire vulnerabilità come SQL Injection. È stato anche dimostrato come gli attacchi possano essere rilevati e analizzati tramite strumenti avanzati di monitoraggio del traffico, sottolineando l'importanza di un attento monitoraggio delle reti e della risposta rapida alle anomalie rilevate.
In conclusione, il laboratorio ha fornito una panoramica pratica sui metodi di rilevamento degli attacchi SQL Injection e sulle tecniche di analisi del traffico utilizzando strumenti professionali, evidenziando la necessità di misure preventive e reattive per proteggere i sistemi da attacchi informatici.