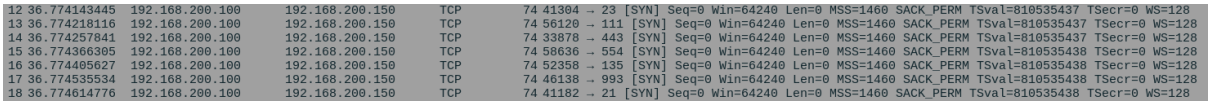
Progetto S9-L5

- ## Obiettivo:

1. Analizzare una cattura di rete data, effettuata con **Wireshark**, al fine di individuare indicatori di compromissione (**IoC**) che indicano la presenza di un attacco in corso.
2. In base agli IoC trovati, fare delle ipotesi su eventuali vettori di attacco utilizzati.
3. Consigliare un'azione per ridurre gli impatti dell'attacco attuale ed eventualmente un simile attacco futuro.

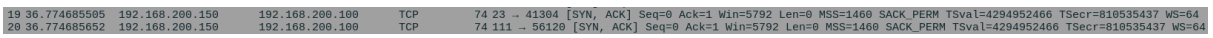
- ## Punto 1: Analisi della cattura di rete

Il primo passo per raggiungere gli obiettivi prefissati è stato quello di analizzare la cattura di rete effettuata tramite Wireshark.
La prima cosa che possiamo notare è una grande quantità di richieste **TCP** con sorgente la macchina con indirizzo IP **192.168.200.100** e destinazione la macchina **192.168.200.150**.
Questo può essere un primo campanello di allarme che ci fa formulare l'ipotesi che la 192.168.200.100 sia la macchina attaccante, mentre la 192.168.200.150, sia la macchina attaccata.
Si tratta di richieste con la flag **SYN**, cioè il primo passo del meccanismo three-way-handshake utilizzato per instaurare comunicazioni TCP.

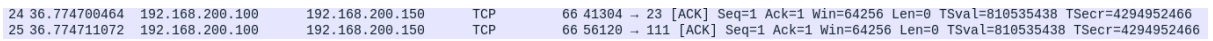


Un'altra cosa che possiamo notare riguarda le porte sorgente e le porte di destinazione, esse infatti cambiano in continuazione.
Il fatto che la porta sorgente cambi ad ogni tentativo di connessione potrebbe rappresentare un modo di eludere alcuni sistemi di sicurezza da parte di una macchina attaccante.
Invece il fatto che la porta di destinazione sia ogni volta diversa, può indicare una scansione in corso da parte della macchina attaccante, la quale sta provando a "bussare" a tutte le porte della macchina 192.168.200.150 per individuare eventuali porte aperte.
In particolare, se la porta analizzata risulterà chiusa, la macchina attaccata risponderà con un **RST** chiudendo la connessione su quella porta.

Se la porta invece risulta essere aperta, la macchina attaccata risponderà con un **SYN-ACK**



Al quale a sua volta la macchina attaccante risponderà con un **ACK** completando il three-way-handshake.



Tale comportamento è tipico di una scansione invasiva effettuata con tool specifici come nmap.
Un esempio del comando lanciato potrebbe essere:

nmap -sT 192.168.200.150

Con l'aggiunta dell'istruzione **-p** nel caso in cui si volesse specificare il range delle porte da analizzare.
Ricapitolando quindi, gli **IoC** che indicano che un attacco è in corso sono:

1. Numerose richieste **SYN** effettuate sul sistema target, con conseguenti risposte **SYN-ACK** e **ACK** in caso di porte aperte, **RST** in caso di porte chiuse.
2. In generale, connessioni multiple in rapida successione su porte diverse.
3. Indirizzo IP malevolo che sta cercando di connettersi alla nostra macchina (**192.168.200.100**).
4. La presenza di numerose porte sorgente differenti in rapida successione.

● Punto 2: Vettori di attacco

Come detto precedentemente, un probabile vettore di attacco utilizzato in questo caso è un tool specifico per la scansione di rete, in grado di effettuare un'analisi invasiva tramite protocollo TCP, come **nmap**.
Tuttavia in questa sezione, voglio anche soffermarmi sui possibili vettori di attacco che un attaccante potrebbe utilizzare in seguito alla scansione effettuata.
Analizzando il file su Wireshark, possiamo capire quali porte aperte sono state individuate dall'attaccante: 21,22,23,25,53,80,111,139,445,512,513,514.
Vediamo quali vettori di attacco potrebbero essere utilizzati su alcune di queste porte.

- **Porta 21 - Servizio FTP**
    1. Attacchi brute force con tool come **Hydra**.
    2. **Accesso anonimo**: FTP può essere configurato per consentire l'accesso anonimo.
    3. Vulnerabilità nel Software FTP: alcuni server FTP (come vsFTPd) hanno vulnerabilità note che possono permettere l'esecuzione di codice remoto.
    4. Attacchi **MITM**: FTP trasmette dati in chiaro, inclusi username e password.

- **Porta 22 - SSH**
    1. Attacchi brute force.
    2. Vulnerabilità nel software SSH: alcune versioni di SSH possono contenere bug o avere vulnerabilità note da sfruttare.
    3. Attacchi **MITM**: se le chiavi SSH non sono verificate correttamente, un attaccante può intercettare la connessione e decriptare la sessione.

- **Porta 23 - Telnet**
    1. Accesso non autenticato: Telnet non è criptato e le credenziali sono trasmesse in chiaro.
    2. Brute force su Telnet.
    3. **Sniffing**: Gli attaccanti possono monitorare il traffico non criptato per catturare informazioni sensibili.
    4. Vulnerabilità presenti nel software.

- **Porta 25 - SMTP**
    1. Server SMTP mal configurati possono permettere l'invio di email senza restrizioni, il che può essere sfruttato per inviare **spam** o **phishing**.
    2. Brute force su SMTP.
    3. **Spoofing**: Gli attaccanti possono falsificare l'indirizzo del mittente in un'email.
    4. **Denial of Service** (**DoS**): attacchi per sovraccaricare il server SMTP e impedirne il corretto funzionamento.
    5. Vulnerabilità presenti nel software.

- **Porta 53 - DNS**
    1. **DNS Amplification Attack** (**DDoS**): utilizza richieste DNS con indirizzi IP falsificati per saturare la rete di una vittima.
    2. **DNS Cache Poisoning**: manipolazione delle risposte DNS per reindirizzare gli utenti verso siti malevoli.
    3. **Esfiltrazione di dati tramite DNS tunneling**: canali nascosti per trasferire dati attraverso richieste DNS.

- **Porta 80 - HTTP**
    1. **Cross-Site Scripting** (**XSS**): gli attaccanti iniettano codice JavaScript maligno nelle pagine web per rubare dati sensibili degli utenti.
    2. **SQL Injection**: gli attaccanti sfruttano le vulnerabilità nelle applicazioni web per eseguire query SQL dannose e ottenere accesso ai database.
    3. **Buffer Overflow**: vulnerabilità nei server web o nelle applicazioni web che permettono l'esecuzione di codice arbitrario.
    4. Attacchi **MITM** grazie all'utilizzo del protocollo HTTP al posto di HTTPS.

- Punto 3: Mitigazione impatto e rischi

Metodi per ridurre l'impatto dell'attacco in corso:

1. Una prima risposta immediata è bloccare l'indirizzo IP dell'attaccante. E' possibile fare ciò tramite un **firewall**, utilizzando regole per bloccare l'IP che sta eseguendo la scansione.
2. Limitare il numero di connessioni che un singolo IP può effettuare verso il tuo server. Questo può ridurre l'effetto di una scansione massiva di porte.
3. Un **WAF** (**Web Application Firewall**) può filtrare il traffico dannoso e impedire che la scansione abbia successo, specialmente se le porte aperte sono legate a servizi web.
4. Chiudere porte non in utilizzo. Ad esempio, se non si sta utilizzando il servizio SSH, la porta 22 può essere chiusa. Ciò può essere fatto o direttamente sul server o tramite firewall.

Metodi per prevenire futuri attacchi:

1. Monitoraggio costante delle attività di rete con strumenti come Wireshark.
2. Implementare un firewall con regole di filtraggio del traffico.
3. Mantenere costantemente aggiornati sistemi, servizi e relative patch di sicurezza.

4. Utilizzare un **IDS/IPS** per monitorare e bloccare automaticamente attacchi come scansioni di porte.
5. Ridurre la superficie di attacco disabilitando i servizi non necessari.
6. Implementare autenticazioni a più fattori sicure per servizi sensibili come SSH (porta 22).
7. Monitorare i log di accesso in modo da rilevare tempestivamente tentativi di scansioni e accessi non autorizzati.
8. Cambiare la porta di default di alcuni servizi, come SSH (porta 22) o MySQL (porta 3306), a una porta meno conosciuta. Ciò può ridurre la probabilità che un attaccante la rilevi facilmente.
9. Per servizi come SSH, limitare l'accesso solo a comunicazioni tramite una **VPN** o **tunnel SSH** per garantire che solo i client autorizzati possano connettersi.
10. **BONUS**: implementare **honeypots** per deviare le scansioni da sistemi reali e raccogliere informazioni sugli attacchi. Un honeypot è un sistema progettato per sembrare vulnerabile ma che è isolato dal resto della rete.
11. **BONUS**: implementare **SIEM** per un monitoraggio centralizzato più efficiente e **SOAR** per l'automazione di meccanismi di difesa.