EXPLOIT

S7.L5

- Traccia:

Esercizio Traccia e requisiti La nostra macchina Metasploitable presenta un servizio vulnerabile sulla porta 1099 - Java RMI. Si richiede allo studente di sfruttare la vulnerabilità con Metasploit al fine di ottenere una sessione di Meterpreter sulla macchina remota. I requisiti dell'esercizio sono:

- La macchina attaccante (KALI) deve avere il seguente indirizzo IP- 192.168.11.111
- La macchina vittima (Metasploitable) deve avere il seguente indirizzo IP- 192.168.11.112
- Una volta ottenuta una sessione remota Meterpreter, lo studente deve raccogliere le seguenti evidenze sulla macchina remota:

  1) configurazione di rete.

  2) informazioni sulla tabella di routing della macchina vittima.

Cosa sono gli exploit?

Gli exploit sono strumenti o tecniche creati per sfruttare le vulnerabilità presenti in sistemi operativi, software, applicazioni o servizi. Quando un software, un'applicazione o un servizio presenta una vulnerabilità, un exploit può essere utilizzato per:

- Rubare dati sensibili,
- Ottenere accesso non autorizzato a un sistema,
- Eseguire del codice malevolo.

Esistono diversi tipi di exploit, classificati in base al tipo di vulnerabilità che sfruttano:

1. Exploit per le Applicazioni Web:

Questi attacchi mirano alle vulnerabilità delle applicazioni web, compromettendone la sicurezza.

Includono:

- SQL Injection: sono attacchi informatici in cui un utente malintenzionato inserisce codice SQL malevolo nei campi di input di un'applicazione web per manipolare il database
- XSS (Cross-Site Scripting): sono vulnerabilità che permettono a un attaccante di iniettare codice malevolo in una pagina web (permettendo azioni come furto di cookie o manipolazione del contenuto della pagina.)
- CSRF (Cross-Site Request Forgery): sono attacchi in cui un attaccante porta un utente autenticato a eseguire azioni non desiderate su un sito web fidato inviando richieste fraudolente.

2. Exploit per le Reti

Questi sfruttano le vulnerabilità nei protocolli o nella configurazione delle reti per intercettare o manipolare il traffico.

Esempi:

- **MITM (Man-in-the-Middle):** Questo avviene quando un attaccante si pone nel mezzo della comunicazione tra due parte, per intercettare, alterare o spiare i dati scambiati. Questo porta spesso al furto di credenziali e dati sensibili sfruttando connessioni in chiaro o vulnerabilità nei protocolli di rete.
- **DNS Spoofing (avvelenamento delle cache) :** è un attacco in cui il black hat modifica le risposte dei server dns e reindirizza gli utenti verso siti falsi (copie esatte degli originali) e ciò permette di rubare dati come le credenziali di accesso o l'invio di malware
- **Packet Sniffing:** Cattura del traffico di rete per analizzare i pacchetti in transito.

3. Exploit per i Sistemi Operativi

- Questi attacchi prendono di mira le vulnerabilità dei sistemi operativi per ottenere accesso non autorizzato

Includono:

- **Escalation dei Privilegi:** L'attaccante guadagna diritti di amministratore su un sistema operativo, ottenendo un controllo completo della macchina.
- **Buffer Overflow:** Scrivere dati oltre i limiti di un buffer (area di memoria temporanea), ciò può causare la sovrascrittura delle aree adiacenti di memoria causando dei malfunzionamenti come il crash del programma o in alcuni casi l'esecuzione del codice malevolo, poiché non essendo in un'unica zona non viene riconosciuto dai sistemi di sicurezza

Exploit Zero-Day

Oltre ai tipi di exploit già mensionati, è importante menzionare gli **exploit zero-day**, che sfruttano vulnerabilità sconosciute o non ancora risolte. Il termine "zero-day" deriva dal fatto che, quando l'exploit viene utilizzato, non esistono ancora aggiornamenti o patch per risolvere quella vulnerabilità, il che significa che il programmatore del software ha "zero giorni" per correggerla. Questi exploit sono tra i più pericolosi, poiché il servizio attaccato è privo di difese conosciute, rendendo più facile per l'attaccante sfruttare la vulnerabilità prima che venga rilevata e corretta.

L'importanza di Conoscere gli Exploit nella Cybersecurity

- Nel campo della cybersecurity, gli exploit rappresentano uno degli strumenti principali utilizzati dagli attaccanti per sfruttare le vulnerabilità dei sistemi informatici. Conoscere gli exploit è fondamentale per:
- **Prevenire attacchi informatici**: Conoscendo i tipi di vulnerabilità presenti nei sistemi e gli exploit che potrebbero essere utilizzati, è possibile adottare misure preventive per proteggere i sistemi da potenziali attacchi.
- **Rilevamento tempestivo**: La conoscenza degli exploit consente di individuare rapidamente un attacco in corso e di rispondere in modo efficace, riducendo al minimo i danni.
- **Sviluppare difese migliori**: Conoscere gli exploit aiuta a sviluppare difese adeguate e a implementare soluzioni di sicurezza mirate per affrontare le vulnerabilità conosciute.
- **Mitigare i rischi**: Una comprensione approfondita degli exploit permette di essere meglio preparati a gestire i rischi legati a vulnerabilità non ancora risolte o sconosciute.

In conclusione conoscere gli exploit è essenziale per qualsiasi professionista della cybersecurity, poiché consente di proteggere meglio i sistemi, rispondere agli attacchi in modo rapido ed efficiente, e ridurre i rischi legati a vulnerabilità sconosciute. La sicurezza informatica non è solo una questione di reagire agli attacchi, ma di essere preparati ad affrontarli proattivamente.

L'Importanza degli Exploit nel PenTesting

Conoscere gli exploit è fondamentale non solo per proteggersi da vari tipi di attacchi informatici, ma anche per condurre efficacemente attività di pentesting. Queste attività sono essenziali per identificare, testare e risolvere le vulnerabilità presenti nei sistemi e nelle applicazioni.

Vediamo ora perché è così importante:

1. Simulazione di Attacchi Realistici: ciò permette di simulare attacchi informatici, riproducendo scenari reali, ciò aiuta a valutare le difese esistenti ed ad individuare possibili falle.

2. Identificazione di Vulnerabilità Nascoste: permette di identificare dei punti deboli nascosti che potrebbero sfuggire durante una scansione superficiale.

3. Verifica dell'Escalation dei Privilegi: conoscere gli exploit consente di testare se un attaccante può ottenere privilegi elevati, come l'accesso root, e fino a che punto può penetrare nel sistema. Questo è fondamentale per valutare la sicurezza complessiva e prevenire accessi non autorizzati.

Raccomandazioni per la Correzione e la Mitigazione: al termine del pentest, il tester fornisce raccomandazioni per correggere le vulnerabilità riscontrate e mitigare i rischi. Questi suggerimenti aiutano a implementare soluzioni di sicurezza efficaci, come aggiornamenti software, modifiche nelle configurazioni di rete e l’introduzione di misure di protezione avanzate dove necessarie.

Passiamo all'esecuzione dell'esercizio!

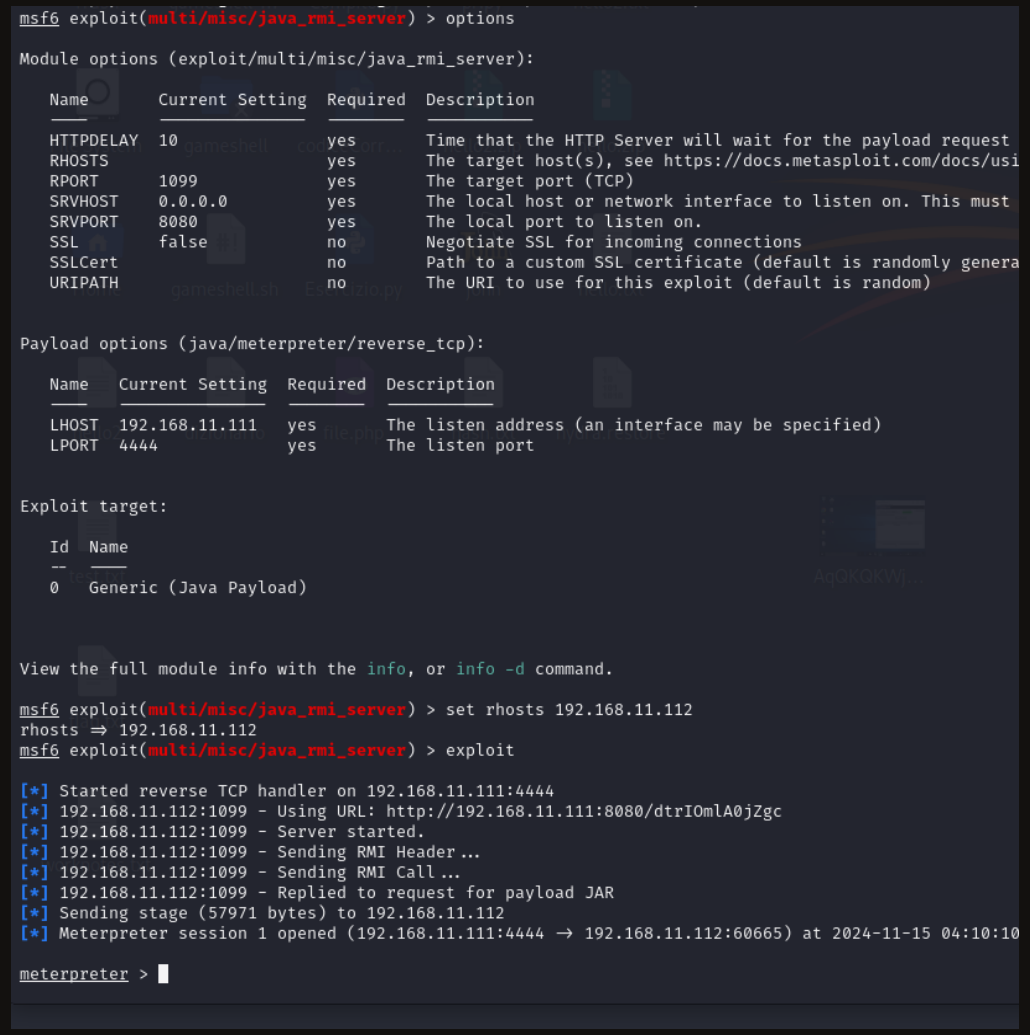
L'esercizio di oggi ci chiede di utilizzare **Metasploit** per sfruttare una vulnerabilità nella porta **1099** di **Java RMI** sulla macchina **Metasploitable**. Come prima cosa, andiamo a configurare gli indirizzi IP come richiesto: sulla macchina attaccante **Kali Linux** impostiamo l'indirizzo IP **192.168.11.111**, mentre sulla macchina target **Metasploitable** impostiamo l'indirizzo IP **192.168.11.112**. Una volta configurati gli IP, iniziamo con l'esercizio vero e proprio.

Anche se non esplicitamente richiesto dall'esercizio, eseguiamo una scansione con **Nmap** per verificare che la porta **1099** sia effettivamente aperta e che il servizio **Java RMI** sia in esecuzione sulla macchina **Metasploitable, successivamente** L'output di **Nmap** ha confermato che la porta **1099** era aperta, con un servizio **Java RMI** in esecuzione.

Confermata la presenza della vulnerabilità, abbiamo avviato **Metasploit** sulla macchina **Kali** utilizzando il comando msfconsole. Successivamente, abbiamo cercato l'exploit da utilizzare tramite la funzione search e abbiamo scelto di utilizzare l'exploit : "**exploit/multi/misc/java_rmi_server"**. A questo punto, abbiamo caricato l'exploit selezionato utilizzando il comando use.

Dopo aver selezionato l'exploit da utilizzare il passo successivo è stato quello di esaminare le opzioni disponibili per la configurazione dell'exploit tramite il comando options.

Questo comando ci ha mostrato le variabili che necessitano di configurazione, come l'indirizzo IP della macchina target (**RHOST**). Abbiamo quindi impostato questo valore, come mostrato nell'immagine della slide successiva, inoltre, abbiamo verificato che gli altri parametri, come la porta di destinazione, fossero correttamente configurati con i valori predefiniti.



Come possiamo vedere dall'immagine, il comando options ci ha mostrato i parametri da configurare, in particolare RHOST (l'indirizzo IP della macchina target). Inoltre, abbiamo potuto verificare che gli altri parametri, come la porta di destinazione (RPORT), erano già correttamente configurati correttamente.

Dopo aver verificato che tutte le informazioni fossero corrette, abbiamo lanciato l'attacco utilizzando il comando: exploit.

E come possiamo vedere il lancio dell'attacco ha avuto successo, e Meterpreter si è avviato correttamente, segnando l'ottenimento della sessione remota sulla macchina Metasploitable.

Dopo aver ottenuto la sessione Meterpreter, siamo in grado di eseguire una serie di comandi per raccogliere informazioni sulla macchina target, per soddisfare i requisiti dell'esercizio, dobbiamo ottenere due tipi di informazioni: la configurazione di rete e la tabella di routing della macchina vittima.

Per raccogliere i dettagli della configurazione di rete, abbiamo eseguito il comando "ifconfig" che ci ha mostrato tutte le informazioni relative agli indirizzi IP, alle interfacce di rete e alla configurazione complessiva della rete sulla macchina compromessa.

Successivamente, per visualizzare la tabella di routing, abbiamo utilizzato il comando "route" che ci ha fornito informazioni sulle rotte di rete configurate sulla macchina vittima.

Come possiamo vedere nell'immagine di fianco, i risultati ottenuti dai comandi mostrano chiaramente la configurazione di rete e la tabella di routing della macchina compromessa.

```
meterpreter > ifconfig

Interface  1
============
Name         : lo - lo
Hardware MAC : 00:00:00:00:00:00
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ::


Interface  2
============
Name         : eth0 - eth0
Hardware MAC : 00:00:00:00:00:00
IPv4 Address : 192.168.11.112
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::a00:27ff:fe23:73ef
IPv6 Netmask : ::

meterpreter > route

IPv4 network routes
===================

    Subnet          Netmask        Gateway  Metric  Interface
    ------          -------        -------  ------  ---------
    127.0.0.1       255.0.0.0      0.0.0.0
    192.168.11.112  255.255.255.0  0.0.0.0


IPv6 network routes
===================

    Subnet                    Netmask  Gateway  Metric  Interface
    ------                    -------  -------  ------  ---------
    ::1                       ::       ::
    fe80::a00:27ff:fe23:73ef  ::       ::
```

HTTPDELAY

Nella traccia dell'esercizio ci viene nominato il parametro **HTTPDELAY**, che avrebbe potuto causare problemi con il lancio dell'exploit se non fosse stato configurato correttamente.

Iniziamo dicendo che **HTTPDELAY** è un parametro utilizzato in Metasploit che serve a controllare il tempo di ritardo tra una richiesta HTTP e l'invio della risposta. Questo parametro consente di introdurre un intervallo di tempo tra le risposte HTTP inviate dal server dell'attaccante, riducendo così la velocità del traffico di rete.

L'utilizzo di **HTTPDELAY** è particolarmente utile quando si cerca di ridurre la possibilità che un attacco venga rilevato dai sistemi di difesa come i firewall.

Se il traffico è troppo rapido o troppo consistente, potrebbe essere individuato come anomalo e bloccato.

Tuttavia, con un appropriato **HTTPDELAY**, l'attacco risulta meno sospetto, aumentando così le probabilità di successo. Inoltre, ciò rende l'attacco più difficile da tracciare, rendendo la rilevazione da parte delle difese della macchina target molto più difficile.

Se non configurato correttamente, **HTTPDELAY** potrebbe causare un errore nel flusso dell'attacco, rallentando troppo la comunicazione o rendendo il traffico troppo evidente.

Conclusione!

In conclusione, questo esercizio ci ha permesso di approfondire il funzionamento degli exploit e di imparare come utilizzarli in modo sicuro e strategico, migliorando le nostre competenze in cybersecurity e penetration testing. Abbiamo appreso come sfruttare vulnerabilità specifiche, come quella del servizio Java RMI su Metasploitable, e come raccogliere informazioni fondamentali sulla macchina vittima, come la configurazione di rete e la tabella di routing. Inoltre, l'esercizio ci ha insegnato l'importanza di parametri come **HTTPDELAY**, che permette di ridurre il rischio di rilevamento durante l'attacco e di aumentare le probabilità di successo, rendendo l'operazione più discreta ed efficace. Questo ci ha aiutato a comprendere come configurare correttamente gli strumenti e sfruttare al meglio le vulnerabilità, migliorando le nostre capacità nella difesa e nella gestione degli attacchi informatici.