REPORT DANIELE NIEDDU – ESERCIZIO 22/11/2024

FINE MODULO

INDICE:

Traccia Esercizio:

Effettuare una scansione completa sul target **Metasploitable**. Scegliete da un minimo di 2 fino ad un massimo di 4 vulnerabilità critiche e provate ad implementare delle azioni di rimedio.
N.B. le azioni di rimedio, in questa fase, potrebbero anche essere delle regole firewall ben configurate in modo da limitare eventualmente le esposizioni dei servizi vulnerabili. Vi consigliamo tuttavia di utilizzare magari questo approccio per non più di una vulnerabilità.
Per dimostrare l'efficacia delle azioni di rimedio, eseguite nuovamente la scansione sul target e confrontate i risultati con quelli precedentemente ottenuti.

Introduzione:

Nella cartella "**M3-REPORT-FINE-MODULO-22-11-24**" troverete altri 2 PDF generati dal programma **NESSUS** (Nessus è uno strumento di **vulnerability assessment** (valutazione delle vulnerabilità), progettato per analizzare reti, sistemi, applicazioni e dispositivi alla ricerca di vulnerabilità, configurazioni errate, e altre potenziali falle di sicurezza.) installato su macchina **Kali Linux** che analizza le vulnerabilità della macchina **Metasploitable2**.

Info Macchine Virtuali

Kali Linux - **IP:** 192.168.1.9

Metasploitable2 - **IP:** 192.168.1.16

Svolgimento Esercizio:

Vulnerabilità CRITICAL Risolte

La scansione iniziale ha riportato **10** vulnerabilità CRITICAL sulla macchina di destinazione, noi ne prenderemo in considerazione **8** qui sotto riportate. Per le vulnerabilità evidenziate di GIALLO son state fatte delle remediation mirate alla risoluzione della singola vulnerabilità, per quanto riguarda invece quelle evidenziate in ROSSO è stata inserita una regola Firewall per bloccare il traffico in ingresso sulle porte interessate impedendo l'accesso remoto ai servizi vulnerabili e quindi mitigando le vulnerabilità senza disattivare i servizi stessi.

51988 - Bind Shell Backdoor Detection

32314 - Debian OpenSSH/OpenSSL Package Random Number Generator Weakness

32321 - Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL check) - porta 5432

32321 - Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL check) - porta 25

20007 - SSL Version 2 and 3 Protocol Detection – porta 5432

20007 - SSL Version 2 and 3 Protocol Detection – porta 25

46882 - UnrealIRCd Backdoor Detection

61708 - VNC Server 'password' Password

Analisi e Remediation Vulnerabilità:

Bind Shell Backdoor Detection

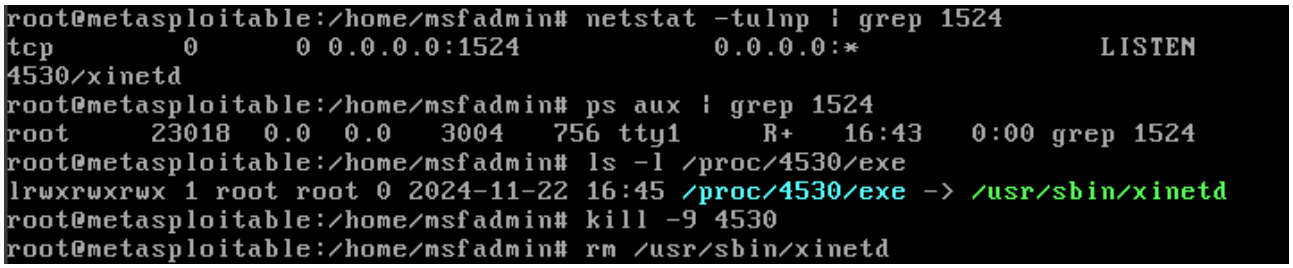
Analisi:

La vulnerabilità Bind Shell Backdoor consente a un attaccante di connettersi direttamente a una porta aperta sul sistema (In questo caso la 1524, ottenuta dal report Nessus), ottenendo accesso remoto non autorizzato. Questa backdoor è spesso installata tramite malware o configurazioni maligne, creando un servizio in ascolto su una porta per eseguire comandi arbitrari.

Risoluzione:

La vulnerabilità è stata risolta identificando il processo attivo associato alla backdoor tramite il comando ps e individuando il file eseguibile nella directory /usr/sbin/xinetd. L'eliminazione dell'intera cartella ha rimosso il servizio compromesso, bloccando la bind shell.

Screenshot:



Spiegazione comandi utilizzati:

- *netstat -tulnp | grep 1524*

  netstat: Mostra informazioni sulle connessioni di rete, sulle porte in ascolto e sui processi associati.
  -t: Visualizza solo le connessioni TCP.
  -u: Visualizza solo le connessioni UDP.
  -l: Mostra solo le porte in ascolto (listening).
  -n: Mostra indirizzi e numeri di porta in formato numerico (senza risoluzione DNS).
  -p: Mostra il processo (PID) e il programma associato a ciascuna porta.
  | grep 1524: Filtra i risultati per mostrare solo le righe che contengono il numero di porta 1524.

- *ps aux | grep 1524*

  ps: Mostra i processi attivi sul sistema.
  a: Mostra i processi di tutti gli utenti.
  u: Mostra i processi in formato dettagliato (incluso utente, CPU, memoria, ecc.).
  x: Mostra anche i processi che non sono legati a un terminale.
  | grep 1524: Filtra i risultati per mostrare solo le righe che contengono 1524.

- *ls -l /proc/4530/exe*

  ls -l: Mostra informazioni dettagliate sui file, inclusi permessi, proprietario, gruppo, dimensioni, e percorso.
  /proc/4530/exe: La directory /proc/ è una rappresentazione del sistema operativo in tempo reale, dove ogni sottodirectory numerica rappresenta un processo attivo (in questo caso, il processo con PID 4530), Il file exe è un collegamento simbolico (symlink) che punta all'eseguibile che il processo sta attualmente utilizzando.

- *kill -9 4530*

  kill: Invoca un segnale per terminare un processo.
  -9: Specifica il segnale SIGKILL, che forza la terminazione immediata del processo senza permettere esecuzioni pulite di cleanup.

- *rm /usr/sbin/xinetd*

  rm: Rimuove un file o una directory.
  /usr/sbin/xinetd: Specifica il percorso del file eseguibile di xinetd

Debian OpenSSH/OpenSSL Package Random Number Generator Weakness

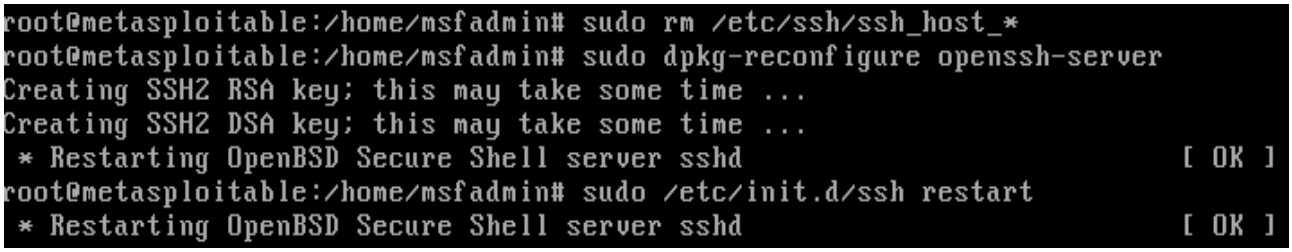
Analisi:

Questa vulnerabilità è causata da una modifica errata nei pacchetti Debian di OpenSSL, che limita la qualità e l'imprevedibilità del generatore di numeri casuali. Ciò rende le chiavi crittografiche SSH prevedibili, esponendo il sistema ad attacchi come brute force o spoofing.

Risoluzione:

La vulnerabilità è stata risolta rimuovendo le chiavi SSH compromesse, rigenerandole con un generatore di numeri casuali corretto e riavviando i servizi SSH per applicare le nuove chiavi.

Screenshot:



Spiegazione comandi utilizzati:

- *sudo rm /etc/ssh/ssh_host_**

  Serve per rimuovere tutte le chiavi host SSH esistenti in quanto compromesse.

- *sudo dpkg-reconfigure openssh-server*

  Rigenera automaticamente le chiavi host SSH e assicura che le nuove chiavi siano sicure e non prevedibili.

- *sudo /etc/init.d/ssh restart*

  Riavvia il servizio SSH per applicare le nuove chiavi e le configurazioni e garantisce che il server utilizzi solo le chiavi rigenerate.

UnrealIRCd Backdoor Detection

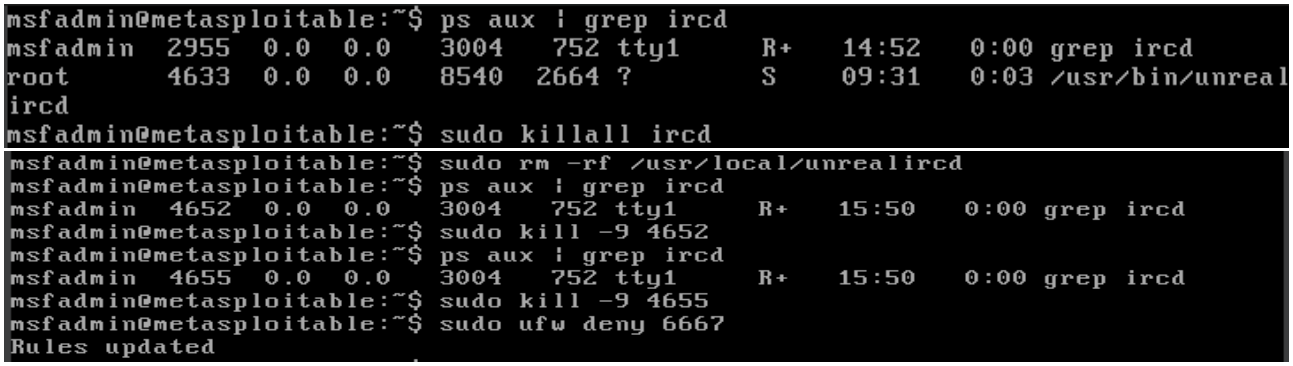
Analisi:

La vulnerabilità UnrealIRCd Backdoor Detection consiste in una backdoor inserita in alcune versioni compromesse di UnrealIRCd, che consente l'esecuzione remota di comandi da parte di un attaccante inviando una stringa appositamente formattata al server IRC.

Risoluzione:

Il problema è stato risolto rimuovendo completamente UnrealIRCd dal sistema, eliminando i file eseguibili e bloccando l'accesso alla porta 6667 utilizzata dal servizio.

Screenshot:



Spiegazione comandi utilizzati:

- *ps aux | grep ircd*

  Mostra tutti i processi attivi (ps aux) e filtra quelli contenenti il termine ircd. Serve per identificare il processo UnrealIRCd in esecuzione, il suo PID (Process ID) e fornisce la directory che contiene l'eseguibile.

- *sudo killall ircd*

  Termina tutti i processi con il nome ircd. Viene usato per chiudere rapidamente UnrealIRCd senza dover cercare manualmente il PID.

- *sudo rm -rf /usr/local/unrealircd*

  Elimina in modo ricorsivo (-r) e forzato (-f) la directory che contiene i file di UnrealIRCd.

- *sudo kill -9 4652*

  Termina in modo forzato (-9) un processo specifico identificato tramite il suo PID.È utile se un processo ircd non risponde al comando killall.

- *sudo ufw deny 6667*

  Configura il firewall UFW (Uncomplicated Firewall) per bloccare l'accesso alla porta 6667, utile a prevenire ulteriori connessioni al servizio IRC anche se il software fosse reinstallato.

VNC Server 'password' Password

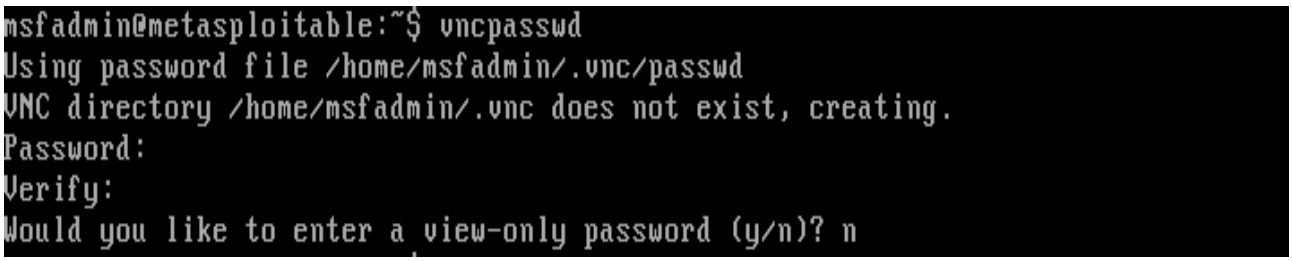
Analisi:

Questa vulnerabilità si verifica quando un server VNC è configurato con la password predefinita "password", facilmente prevedibile e utilizzabile da un attaccante per accedere al sistema in modo non autorizzato.

Risoluzione:

La vulnerabilità è stata risolta sostituendo la password predefinita con una password personalizzata e sicura**.**

Screenshot:



Spiegazione comandi utilizzati:

- *vncpasswd*

  Questo comando permette di configurare o aggiornare la password utilizzata dal server VNC per l'accesso. Una volta eseguito, chiede di inserire e confermare una nuova password. Se configurato, può anche impostare una password "view-only" (che consente solo la visualizzazione senza controllo remoto).

Ora vedremo invece le 4 vulnerabilità evidenziate in ROSSO che son state risolte semplicemente applicando una regola firewall per le porte interessate, ovvero la 5432 e la 25.

Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL check) - porta 5432

Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL check) - porta 25

SSL Version 2 and 3 Protocol Detection – porta 5432

SSL Version 2 and 3 Protocol Detection – porta 25

Analisi:

- *Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL check)*

Questa vulnerabilità, presente su sistemi Debian, deriva da un generatore di numeri casuali difettoso che produce chiavi crittografiche prevedibili. Ciò rende le connessioni SSL/TLS insicure e suscettibili a intercettazioni o spoofing.

- *SSL Version 2 and 3 Protocol Detection*

SSLv2 e SSLv3 sono protocolli obsoleti con gravi vulnerabilità, come l'attacco POODLE, che consentono di intercettare o modificare i dati trasmessi.

I servizi che li supportano, come PostgreSQL che utilizza la porta 5432 per connessioni SSL crittografate e la porta 25 invece utilizzata per il protocollo SMTP, crittografato tramite SSL per l'invio sicuro di email, risultano a rischio.

Risoluzione:

Le vulnerabilità sono state mitigate configurando il firewall per bloccare il traffico in ingresso sulle porte **25** e **5432**, impedendo l'accesso remoto ai servizi vulnerabili e quindi mitigando le vulnerabilità senza disattivare i servizi stessi**.**

Screenshot:

Spiegazione comandi utilizzati:

- *iptables -A INPUT -p tcp --dport 25 -j REJECT*

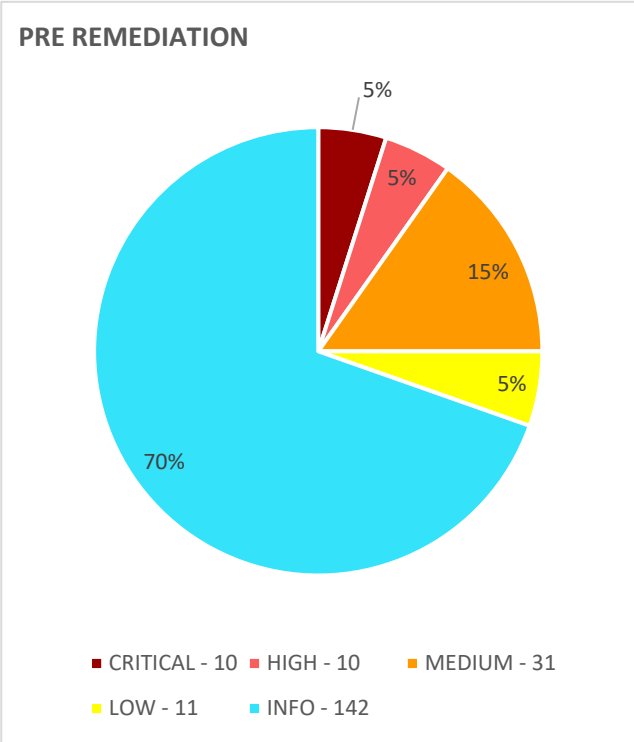
  -A INPUT: Aggiunge una regola alla catena INPUT (traffico in ingresso).
  -p tcp: Specifica che la regola si applica al protocollo TCP.
  --dport 25: Applica la regola al traffico diretto alla porta 25 (SMTP).
  -j REJECT: Blocca attivamente il traffico, informando il mittente che la connessione è stata rifiutata.

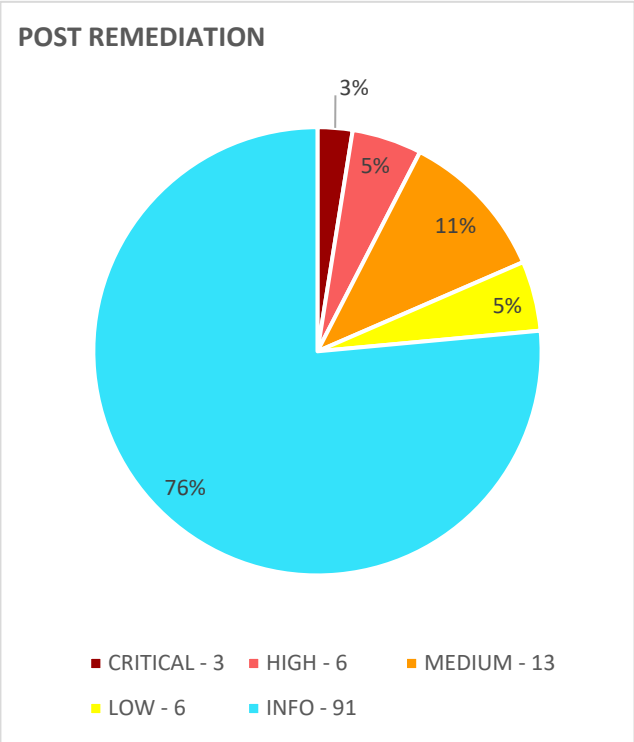
- iptables -A INPUT -p tcp --dport 5432 -j REJECT

  Come il comando precedente, ma applicato alla porta 5432 (PostgreSQL).

Grafici Pre e Post Remediation e Considerazioni finali:

Grafici:





Considerazioni Finali:

La risoluzione delle vulnerabilità in un ambiente IT è cruciale per garantire la sicurezza del sistema, prevenire accessi non autorizzati e difendersi da potenziali attacchi informatici. Intervenire su queste problematiche consente di rafforzare la resilienza complessiva delle infrastrutture, proteggendo i dati sensibili e assicurando la continuità operativa dei servizi.

Come evidenziato dai grafici sopra, le azioni intraprese hanno portato a una significativa riduzione delle criticità nel sistema, abbassando non solo le vulnerabilità di tipo critico, ma anche quelle di gravità inferiore. Per alcune problematiche non completamente risolte, come phpMyAdmin e Apache PHP-CGI Remote Code Execution, sarebbe stato sufficiente un aggiornamento alla versione più recente, in cui il produttore ha già affrontato e risolto tali vulnerabilità.