Svolgimento:

- Scansione iniziale dove si vedono tutte le vulnerabilità e le vulnerabilità da risolvere (**ScansioneInizio.pdf**).
- Screenshot e spiegazione dei passaggi della remediation (**RemediationMeta.pdf**)
- Scansione dopo le modifiche che evidenzia la risoluzione dei problemi/vulnerabilità e mostre le vulnerabilità ancora presenti) **ScansioneFine.pdf**.

Nessus è un applicativo di Vulnerability Assessment che fa la scansione sul client Metasploitable (target) per rilevare le criticità e implementare azioni di rimedio per quattro di queste vulnerabilità critiche/high.

Scansione iniziale:

192.168.50.101

8	4	16	6	73
CRITICAL	HIGH	MEDIUM	LOW	INFO

Vulnerabilities — Total: 107

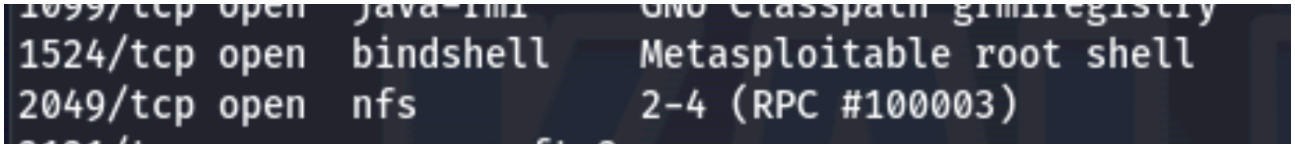
SEVERITY	CVSS V3.0	VPR SCORE	PLUGIN	NAME
CRITICAL	9.8	9.0	134862	Apache Tomcat AJP Connector Request Injection (Ghostcat)
CRITICAL	9.8	-	51988	Bind Shell Backdoor Detection
CRITICAL	9.8	-	20007	SSL Version 2 and 3 Protocol Detection
CRITICAL	10.0	-	33850	Unix Operating System Unsupported Version Detection
CRITICAL	10.0*	7.4	32314	Debian OpenSSH/OpenSSL Package Random Number Generator Weakness
CRITICAL	10.0*	7.4	32321	Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL check)
CRITICAL	10.0*	5.9	11356	NFS Exported Share Information Disclosure
CRITICAL	10.0*	-	61708	VNC Server 'password' Password
HIGH	8.6	5.2	136769	ISC BIND Service Downgrade / Reflected DoS
HIGH	7.5	-	42256	NFS Shares World Readable
HIGH	7.5	6.1	42873	SSL Medium Strength Cipher Suites Supported (SWEET32)
HIGH	7.5	6.7	90509	Samba Badlock Vulnerability

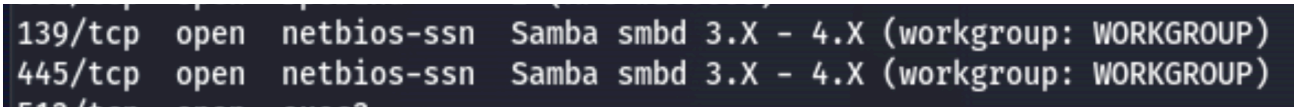
Le 4 vulnerabilità scelte sono:

CRITICAL	9.8	-	51988	Bind Shell Backdoor Detection
CRITICAL	10.0*	5.9	11356	NFS Exported Share Information Disclosure
CRITICAL	10.0*	-	61708	VNC Server 'password' Password
HIGH	7.5	6.7	90509	Samba Badlock Vulnerability

È utile inizialmente eseguire su Kali Linux una scansione delle porte e dei servizi in ascolto con il comando: nmap -sV 192.168.50.101

In questo modo notiamo le porte 1524, 139 e 445:





11356 - NFS Exported Share Information Disclosure

La prima vulnerabilità è NFS Exported Share Information Disclosure.

Introduzione a NFS
NFS, acronimo di Network File System, è un protocollo di rete progettato specificamente per la condivisione di file e directory attraverso una rete di computer. Consente a diversi dispositivi e sistemi sulla rete di accedere e condividere file e risorse memorizzate su un server remoto come se fossero memorizzate localmente sul proprio sistema.
NFS opera utilizzando un sistema client-server, dove il server esporta (rendere disponibili) determinate directory o file, e i client montano (collegano) queste risorse remote sul proprio sistema per accedervi.
NFS è ampiamente utilizzato in ambienti Unix e Linux per la condivisione di risorse tra sistemi, facilitando la collaborazione e l'accesso centralizzato ai file.

Quando un dispositivo accede a una condivisione NFS, deve fornire le proprie credenziali di accesso. Queste credenziali vengono utilizzate per verificare se l'utente ha il permesso di accedere alla condivisione.

Vulnerabilità
La vulnerabilità NFS è un bug nel modo in cui il protocollo NFS gestisce le autorizzazioni quindi il problema riguarda la configurazione delle condivisioni NFS su un server remoto.
La vulnerabilità NFS può essere usata da un utente malintenzionato quando invia un messaggio NFS appositamente predisposto al server NFS e questo messaggio sfrutta il bug. Il risultato è che l'utente malintenzionato può sfruttare questa configurazione per accedere ai file sul server remoto a cui non dovrebbe avere accesso, anche se non ha le credenziali appropriate.
Il problema potrebbe quindi essere l'esposizione non autorizzata di informazioni sensibili tramite condivisioni NFS (Network File System).

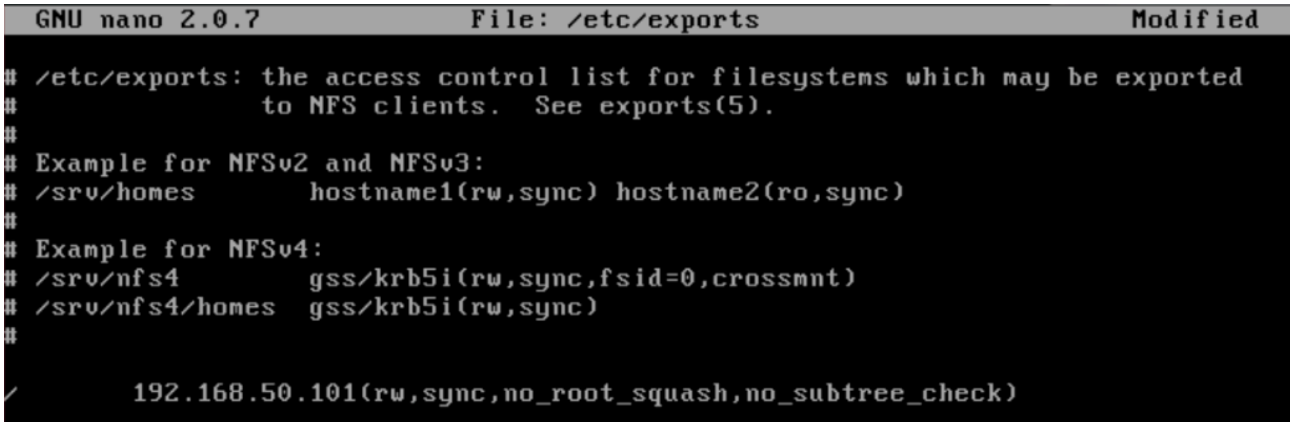
Consigli per configurare le condivisioni NFS in modo sicuro: - Utilizzare le autorizzazioni più restrittive possibili; - Limitare l'accesso alle condivisioni solo agli utenti autorizzati; - Utilizzare un firewall per bloccare l'accesso alle condivisioni da parte di utenti non autorizzati.

In breve:
L'host che esegue la scansione potrebbe essere in grado di connettersi ed accedere ad almeno una condivisione NFS resa disponibile dal server remoto durante il processo di scansione. Un attaccante potrebbe leggere o anche scrivere file sull'host remoto.

Soluzione
Nessus consiglia di configurare NSF sull'host remoto in modo che solo gli host autorizzati possano montare connessioni remote e quindi accedere alle condivisioni NFS. Ho deciso di modificare la configurazione di NFS in questo modo.

Ecco i passaggi che ho usato per risolvere la vulnerabilità NFS:

1. Aprire il file /etc/exports sul server NFS per modificarlo
2. Aggiungere l'indirizzo IP di Metasploitable alla lista degli host autorizzati: la riga aggiunta indica che la condivisione può essere montata solo dall'host con indirizzo IP 192.168.50.101.



Andiamo a rendere le share directory del NFS non accessibile ad altri client, nello specifico i privilegi di read and write.

3. Salva le modifiche e chiudi il file
4. Riavvia il servizio NFS

In alternativa, si dovrà andare con Meta a cambiare un parametro nella configurazione di NFS -> è possibile commentare la riga che esporta la condivisione NFS. Per fare ciò, è necessario aggiungere il simbolo # all'inizio dell'ultima riga (/share).

51988 - Bind Shell BackDoor Detection

La seconda vulnerabilità è Bind Shell BackDoor Detection.

La "Bind Shell Backdoor Detection" si riferisce alla ricerca di una particolare forma di backdoor, conosciuta come "Bind Shell", all'interno di un sistema o di una rete.
La "Bind Shell" è una tipologia di backdoor che apre una porta su un sistema e si mette in ascolto (in "bind") di connessioni in ingresso. Questo tipo di backdoor è progettato per stabilire una connessione di rete che permette all'attaccante di ottenere un accesso remoto al sistema compromesso.

Vulnerabilità
Questa backdoor potrebbe essere utilizzata da un utente malintenzionato per ottenere un accesso non autorizzato perchè potrebbe collegarsi alla porta remota e prendere il controllo per eseguire comandi.
Quindi il problema è che una shell è in ascolto sulla porta remota senza chiedere prima l'autenticazione.

Soluzione
Nessus suggerisce di verificare se l'host è stato manomesso da remoto e, se necessario, reinstallare il sistema.

Ho deciso di:
- Verificare che l'host non sia stato compromesso
- Creare e aggiungere una regola nel firewall per bloccare il traffico su una specifica porta, bloccando le connessioni alla porta che utilizza il servizio quando non è utilizzato. Nel caso in cui il servizio sia invece utilizzato è opportuno aggiungere un'autenticazione all' accesso remoto.

Quindi andremo a configurare un Firewall utilizzando i comandi:
UFW ENABLE: questo comando attiva il firewall sul sistema. Quando il firewall è abilitato, inizierà a monitorare il traffico in entrata e in uscita del sistema. Questo comando può essere utilizzato solo se il firewall è disabilitato.
UFW DEFAULT ALLOW:
Quando si imposta la politica predefinita di UFW (Uncomplicated Firewall) su **allow**, si sta essenzialmente stabilendo che tutte le connessioni in entrata non regolate da regole specifiche

devono essere consentite. Questo comando consente tutte le connessioni di base e la possibilità di impostare la politica predefinita del firewall per permettere il traffico non regolamentato.
In altre parole, se il firewall riceve una connessione per la quale non è stata specificata alcuna regola, la consente per impostazione predefinita. Questo comando può essere utilizzato solo se il firewall è abilitato.

Dalla scansione iniziale abbiamo visto che la porta interessata è la 1524, quindi procediamo ad abilitare il firewall e a chiuderla: **UFW DENY 1524**.
Questo comando aggiunge una regola al firewall per bloccare il traffico sulla porta 1524. In altre parole, qualsiasi connessione che proviene o va verso la porta 1524 verrà bloccata dal firewall. Questo serve per chiudere la porta interessata che è responsabile della backdoor.

Alla fine quindi la vulnerabilità è risolta perchè il firewall inizierà a bloccare il traffico sulla porta 1524.

61708 - VNC Server ‘password’ Password

La terza vulnerabilità è VNC Server ‘password’ Password.

Introduzione a VNC
Un server VNC (Virtuale Network Computing) è un sistema software che consente di controllare a distanza un computer tramite un'interfaccia grafica. È quindi un servizio che permette la visualizzazione e il controllo grafico remoto del desktop del sistema. VNC consente all’utente di connettersi al server VNC da un’altra macchina e interagire con l’ambiente desktop in modo simile a come farebbe localmente. Alcuni server VNC vengono installati con una password di default o configurati con password deboli, come appunto la parola "password".

Su Metasploitable il server VNC potrebbe essere configurato per fornire un’interfaccia grafica remota su una determinata porta.

Vulnerabilità
La password ‘password’ è troppo debole e facilmente indovinabile. Il server VNC in esecuzione sull'host remoto è protetto da una password non abbastanza forte. Nessus è riuscito ad effettuare l'accesso utilizzando l'autenticazione VNC e una password ‘password'. Un utente malintenzionato remoto e non autenticato potrebbe quindi sfruttare questo exploit per prendere il controllo del sistema.

Soluzione
Nessus consiglia di proteggete il servizio VNC con una password forte. Ho quindi risolto andando a cambiare la password con una più complessa e sicura:
Accedendo alla directory, ho usato il comando «vncpasswd» sulla macchina Metasploitable per modificare la password e ne ho inserita una più forte.
Inoltre, è buona prassi configurare l'accesso sicuro come l'autenticazione a due fattori, limitando l'accesso solo agli indirizzi IP autorizzati e aggiornare regolarmente il software VNC per correggere eventuali vulnerabilità note.

90509 - Samba Badlock Vulnerability

La quarta vulnerabilità è Samba Badlock Vulnerability

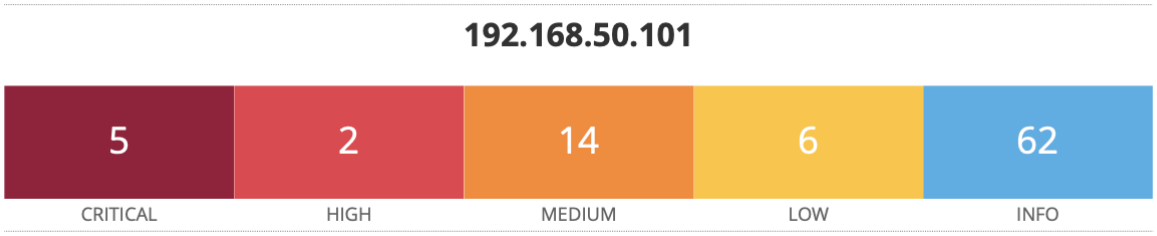
Introduzione a VNC
La “Samba Badlock” si riferisce a una serie di vulnerabilità di sicurezza scoperte nel protocollo di condivisione di file Samba. Samba è un software open-source che implementa il protocollo SMB (Server Message Block)/CIFS(Common Internet File System), consentendo la condivisione di risorse, come file e stampanti, su una rete e la comunicazione tra sistemi Microsoft Windows e UNIX/Linux.

Vulnerabilità
La versione di Samba, in esecuzione sull'host remoto è affetta da un difetto, noto come Badlock, presente nel SAM (Security Account Manager) e nella Local Security Authority (LSA) Domain Policy a causa di una negoziazione impropria del livello di autenticazione sui canali RPC (Remote Procedure Call).
Un attaccante potrebbe sfruttare questa vulnerabilità per ottenere informazioni riservate, modificare i dati nel traffico di rete Samba o eseguire attacchi di tipo “Man-in-the-Middle” per intercettare il traffico tra client e server che ospita un database SAM (Security Account Manager) in un contesto di sistema Windows.
Un MITM è un tipo di attacco in cui un aggressore si inserisce tra la comunicazione tra due parti legittime (in questo caso, il client e il server del database SAM). L'aggressore può intercettare, manipolare o riassemblare il traffico di rete tra le due parti, potenzialmente senza che le parti legittime lo rilevino. L’attaccante potrebbe poi sfruttare la falla per forzare un abbassamento (downgrade) del livello di autenticazione durante la comunicazione tra il client e il server, permettendo all'attaccante di eseguire operazioni non autorizzate nel contesto dell'utente intercettato, con potenziale impatto sulla sicurezza dei dati e dei servizi nel contesto di un ambiente basato su Active Directory.

In breve, questa vulnerabilità era legata a una debolezza nel sistema di autenticazione di Samba, che poteva essere sfruttata da un attaccante per eseguire un attacco di tipo man-in-the-middle, cioè per intercettare e manipolare il traffico di rete tra client e server Samba. Ciò potrebbe permettere ad un utente malintenzionato di ottenere informazioni sensibili o di eseguire azioni non autorizzate all'interno della rete compromessa.

Soluzione
Nessus consiglia di aggiornare alla versione Samba 4.2.11 / 4.3.8 / 4.4.2 o successiva.
Per risolvere questo problema potremmo mettere online la macchina di Metasploitable e aggiornare Samba oppure disabilitare le porte che il servizio utilizza.
Se il servizio non è utilizzato, chiudiamo quindi le porta in ascolto per quel servizio (la 139 e la 445). In questo modo riusciamo a risolvere la vulnerabilità.

Scansione finale:



Vulnerabilities Total: 89

SEVERITY	CVSS V3.0	VPR SCORE	PLUGIN	NAME
CRITICAL	9.8	9.0	134862	Apache Tomcat AJP Connector Request Injection (Ghostcat)
CRITICAL	9.8	-	20007	SSL Version 2 and 3 Protocol Detection
CRITICAL	10.0	-	33850	Unix Operating System Unsupported Version Detection
CRITICAL	10.0*	7.4	32314	Debian OpenSSH/OpenSSL Package Random Number Generator Weakness
CRITICAL	10.0*	7.4	32321	Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL check)
HIGH	8.6	5.2	136769	ISC BIND Service Downgrade / Reflected DoS
HIGH	7.5	6.1	42873	SSL Medium Strength Cipher Suites Supported (SWEET32)

Notiamo che nella scansione finale non sono più presenti le 4 vulnerabilità (3 critical, 1 high) affrontate.

Noto inoltre che non è più presente:
- 42256 NFS Shares World Readable (High)