UNIVERSITÀ DEGLI STUDI DI SALERNO

Dipartimento di Informatica

Corso di Laurea Magistrale in Informatica

Curriculum Software Engineering and IT Management

PENETRATION TESTING AND ETHICAL HACKING

WEB MACHINE: (N7)

Report

DOCENTE

Prof.

Arcangelo Castiglione

STUDENTE

Vincenzo Esposito

Matricola: **0522501385**

Anno Accademico 2022-2023

Sommario

Questo progetto mira a effettuare un processo di Penetration Testing etico utilizzando la macchina vulnerabile "Web Machine: (N7)". L'attività include diverse fasi, come Target Scoping, Information Gathering, Target Discovery, Enumeration Target, Port Scanning, Vulnerability Mapping, Target Exploitation e PostExploitation.

Il progetto prevede l'emulazione di due macchine virtuali, una attaccante e una vittima, utilizzando il software di virtualizzazione UTM. La macchina attaccante è Kali Linux versione Linux 5.17.0-kali3-amd64, mentre la macchina bersaglio è Web Machine: (N7). Le due macchine virtuali sono connesse attraverso una rete locale virtuale con NAT sulla piattaforma di virtualizzazione UTM, utilizzando uno spazio di indirizzi di 192.168.64.0/24. È importante notare che l'indirizzo IP assegnato alla macchina virtuale denominata "Web Machine: (N7)" non è predefinito ma viene stabilito dinamicamente tramite il protocollo DHCP.

Lo scopo del progetto è testare la sicurezza di Web Machine: (N7) utilizzando tecniche di Penetration Testing etico, che prevederanno l'individuazione di eventuali vulnerabilità e la proposta di soluzioni per affrontarle. Il presente progetto avrà l'obiettivo di condurre una valutazione sulla qualità della sicurezza del sistema in esame, allo scopo di identificare eventuali criticità e proporre raccomandazioni per il miglioramento complessivo della sicurezza del sistema.

Indice

Elenco delle figure

CAPITOLO 1

Executive Summary

Il presente progetto mira a effettuare un processo di Penetration Testing etico utilizzando la macchina vulnerabile "Web Machine: (N7)", reperibile al seguente link [1]
Gli obiettivi dell'attività includono:

- Enumerare servizi e vulnerabilità presenti sulla macchina target.
- Sfruttare le vulnerabilità per prendere possesso della macchina target;
- Ottenere la CTF;
- Instaurare un'eventuale Back-Door.

La presente tipologia di attacco si inserisce all'interno della categoria del grey box testing, in quanto, prima di avviare tale procedura, eravamo soltanto a conoscenza del sistema operativo presente sulla macchina di destinazione. Informazioni rilevanti quali l'indirizzo IP e i servizi attivi risultavano sconosciuti. Nella fase di penetration testing, si seguirà altresì l'approccio etico di un hacker di tipo white-hat, con l'obiettivo di scoprire, verificare e segnalare le vulnerabilità del sistema al fine di attestarne la sua fragilità, il tutto nel rispetto delle norme etiche. Si cercherà inoltre di proporre soluzioni atte a mitigare eventuali problemi di sicurezza individuati. Il presente rapporto illustrerà tutte le vulnerabilità che sono state identificate durante il processo di penetration testing.

[1] https://www.vulnhub.com/entry/web-machine-n7,756/

CAPITOLO 2

Engagement Highlights

L'attività di Penetration Testing che è stata condotta ha un obiettivo di natura didattica, pertanto non è stata stabilita alcuna forma di contrattazione con il cliente. Saranno impiegati gli strumenti considerati più efficaci per l'acquisizione delle informazioni e l'esecuzione delle operazioni, senza alcuna restrizione specifica. L'intero progetto ha seguito le fasi apprese nel corso di Penetration Testing and Ethical Hacking:

- Information Gathering & Target Discovery
- Enumeration Target & Port Scanning
- Vulnerability Mapping
- Target Exploitation
- Post-Exploitation (Privilege Escalation & Mantaining Access)

Nella prima fase di Information Gathering & Target Discovery, è stato utilizzato come punto di riferimento lo strumento Nmap, il quale ha consentito di condurre una scansione di rete al fine di identificare l'indirizzo IP della macchina target. Successivamente, per il rilevamento del sistema operativo della macchina target, sono stati impiegati strumenti con funzionalità di fingerprinting del sistema operativo, sia in modalità passiva che attiva, al fine di determinare il sistema operativo del dispositivo preso di mira.

Nella fase di Enumeration Target & Port Scanning, una volta verificata l'esistenza e l'accessibilità della macchina virtuale denominata "Web Machine: (N7)", è stata condotta

un'indagine mirata alla scoperta delle informazioni relative alle porte attive e ai servizi offerti dalla stessa. Attraverso l'enumerazione attiva del target, è stato eseguito un processo di Port Scanning mediante l'utilizzo del tool Nmap, che ha permesso di identificare i servizi e le versioni attive su tutte le porte TCP del sistema di destinazione. Per quanto riguarda le porte UDP, è stato utilizzato il tool Unicornscan, ma non sono state rilevate porte UDP aperte. Per l'enumerazione del sistema, è stato impiegato il tool Dirbuster al fine di individuare eventuali directory indicizzate all'interno del sistema e ottenere informazioni sul servizio offerto dalla macchina target.

Nella fase di Vulnerability Mapping, è stata condotta un'analisi delle vulnerabilità presenti sulla macchina target, sia in modo manuale che automatico. Per l'analisi manuale, è stata effettuata una ricerca sul sito "https://www.exploit-db.com/" per il servizio HTTP sulla porta TCP 80, ma non sono state trovate corrispondenze. Successivamente, è stato utilizzato il tool Vulscan in combinazione con Nmap, che ha evidenziato numerose vulnerabilità che saranno successivamente verificate. A causa del tempo considerevole richiesto per l'analisi manuale, si è optato per l'adozione dell'analisi automatica delle vulnerabilità. È stato impiegato il tool Nessus per eseguire scansioni di rete di base e test delle applicazioni web, poiché la macchina target presenta un'applicazione web. Al fine di approfondire l'analisi delle vulnerabilità legate alle applicazioni web, è stato utilizzato il tool OWASP ZAP per eseguire una scansione automatizzata, selezionando diversi percorsi di scansione. Purtroppo, a causa di problemi di aggiornamento del software sulla macchina del Penetration Tester, non è stato possibile effettuare la scansione con tale strumento durante questa fase.

Nella fase di Target Exploitation, si è intrapreso il tentativo di sfruttare tutte le vulnerabilità presenti sulla macchina target al fine di ottenere vantaggi. Sfruttando appieno le potenzialità offerte da tali vulnerabilità, è stato possibile, attraverso un attacco di Cross-Site Request Forgery e SQL Injection, acquisire le credenziali di accesso della macchina target. Tuttavia, non è stato possibile ottenere l'accesso e il controllo completo della macchina tramite il caricamento di una reverse shell nel form presente nella pagina "http://192.167.64.7/exploit.html".

Le fasi di Post Exploitation non sono state eseguite a causa dell'impossibilità di caricare sul server Apache, dovuta alla mancanza di componenti lasciati aperti dal creatore della macchina virtuale.

CAPITOLO 3

Vulnerability Report

L'analisi condotta sulla macchina virtuale Web Machine: (N7) ha rivelato la presenza di diverse tipologie di vulnerabilità, che saranno illustrate e approfondite in seguito. Tra le vulnerabilità più rilevanti individuate, si evidenziano le seguenti:

- Mancanza di controlli con possibilità di effettuare un attacco di SQL Injection: È stata riscontrata la mancanza di adeguati controlli e validazioni dei dati in ingresso, consentendo potenziali attacchi di SQL Injection. Tale vulnerabilità può compromettere l'integrità e la sicurezza dei dati presenti nel database, aprendo la porta a operazioni non autorizzate o dannose.

- Presenza di commenti lasciati nel codice sorgente dallo sviluppatore, permettendo un attacco di Cross-Site Request Forgery (CSRF): Sono stati individuati commenti nel codice sorgente che possono essere sfruttati da un attaccante per eseguire un attacco di Cross-Site Request Forgery. Questo tipo di attacco mira a ingannare l'utente e indurlo ad eseguire azioni non volute o non autorizzate all'interno dell'applicazione web.

- Presenza di software obsoleti sulla macchina: Sono stati individuati software obsoleti sulla macchina, i quali espongono il sistema a numerose vulnerabilità, sia note che sconosciute. Questo scenario potrebbe consentire a un potenziale utente malintenzionato di ottenere il controllo fisico della macchina, sfruttando tali vulnerabilità.

L'individuazione di queste vulnerabilità sottolinea la necessità di adottare adeguate misure di protezione e di aderire alle migliori pratiche di sviluppo e gestione della sicurezza delle

applicazioni. Saranno fornite ulteriori dettagli sulle vulnerabilità identificate nel proseguo della relazione.

CAPITOLO 4

Remediation Report

La macchina virtuale Web Machine: (N7) presenta un livello di rischio considerabile, per mitigare tale fattore è opportuno adottare misure precauzionali, tra cui:

- Aggiornamento di Apache alla versione 2.4.53 o successiva: È consigliabile effettuare l'aggiornamento del software Apache alla versione 2.4.53 o una versione più recente. Questo permette di beneficiare delle correzioni di bug, delle migliorie di sicurezza e delle patch di vulnerabilità fornite dagli sviluppatori.

- Utilizzo di algoritmi di cifratura sicuri per i cookie, come l'hashing: È opportuno adottare algoritmi di cifratura robusti e sicuri per proteggere i cookie. L'utilizzo dell'hashing rappresenta una buona pratica, in quanto impedisce la decodifica dei dati sensibili e riduce il rischio di furti di sessione o manipolazioni dei cookie.

- Implementazione di controlli adeguati per prevenire attacchi di SQL Injection: È fondamentale adottare misure di sicurezza adeguate, come l'utilizzo di prepared statements o stored procedures, per prevenire attacchi di SQL Injection. Verificare e filtrare attentamente i dati in ingresso, garantendo che siano correttamente sanitizzati e validati.

- Miglioramento delle pratiche di sviluppo, con particolare attenzione alla gestione dei file di test sul server e alla presenza di file che potrebbero consentire modifiche eseguibili da parte di utenti non autorizzati. È essenziale educare i programmatori sulle best practice di sicurezza e sensibilizzarli sui potenziali rischi associati a tali pratiche.

- Pianificazione periodica di controlli di sicurezza: È consigliabile pianificare regolari attività di controllo della sicurezza per valutare l'efficacia delle misure adottate, individuare nuove vulnerabilità e prendere le dovute contromisure per mitigare tali rischi.

- Effettuare aggiornamenti periodici per tutti i servizi attivi sulla macchina: È importante mantenere aggiornati tutti i servizi attivi sulla macchina, compresi il sistema operativo, il software di rete e le applicazioni web. Gli aggiornamenti regolari consentono di beneficiare delle correzioni di sicurezza e delle patch per le vulnerabilità note, riducendo così l'esposizione ai rischi di sicurezza.

L'implementazione di tali contromisure contribuirà a migliorare la sicurezza complessiva della macchina virtuale Web Machine: (N7) e a ridurre il rischio di exploit delle vulnerabilità individuate.

CAPITOLO 5

Findings Summary

Durante l'attività di penetration testing condotta sulla macchina target "Web Machine: (N7)", sono state rilevate numerose vulnerabilità. Tali vulnerabilità sono state classificate in base alla loro gravità in quattro categorie distinte:

- CRITICAL: Questa categoria comprende le vulnerabilità che presentano un impatto significativamente elevato e che possono consentire a un utente malevolo di ottenere un controllo completo o parziale del sistema.
- HIGH: Le vulnerabilità classificate come "HIGH" richiedono particolari requisiti per essere sfruttate e possono avere un impatto relativamente alto sul sistema.
- MEDIUM: Le vulnerabilità classificate come "MEDIUM" sono di complessa sfruttamento e, nella maggior parte dei casi, non presentano un impatto diretto molto significativo sul sistema.
- LOW: Le vulnerabilità classificate come "LOW" hanno un impatto poco significativo e presentano una bassa probabilità di essere sfruttate. Pertanto, non costituiscono una minaccia rilevante per il sistema nel breve termine.

Inoltre, sono state identificate anche informazioni sulla configurazione del software che potrebbero potenzialmente generare vulnerabilità in futuro. Queste informazioni, sebbene non siano considerate vulnerabilità effettive, sono rilevanti per la valutazione della sicurezza complessiva del sistema.

La Figura sottostante illustra il numero di vulnerabilità identificate per ciascuna categoria e un diagramma a torta al fine di fornire una rappresentazione visiva e dettagliata del numero di vulnerabilità individuate:

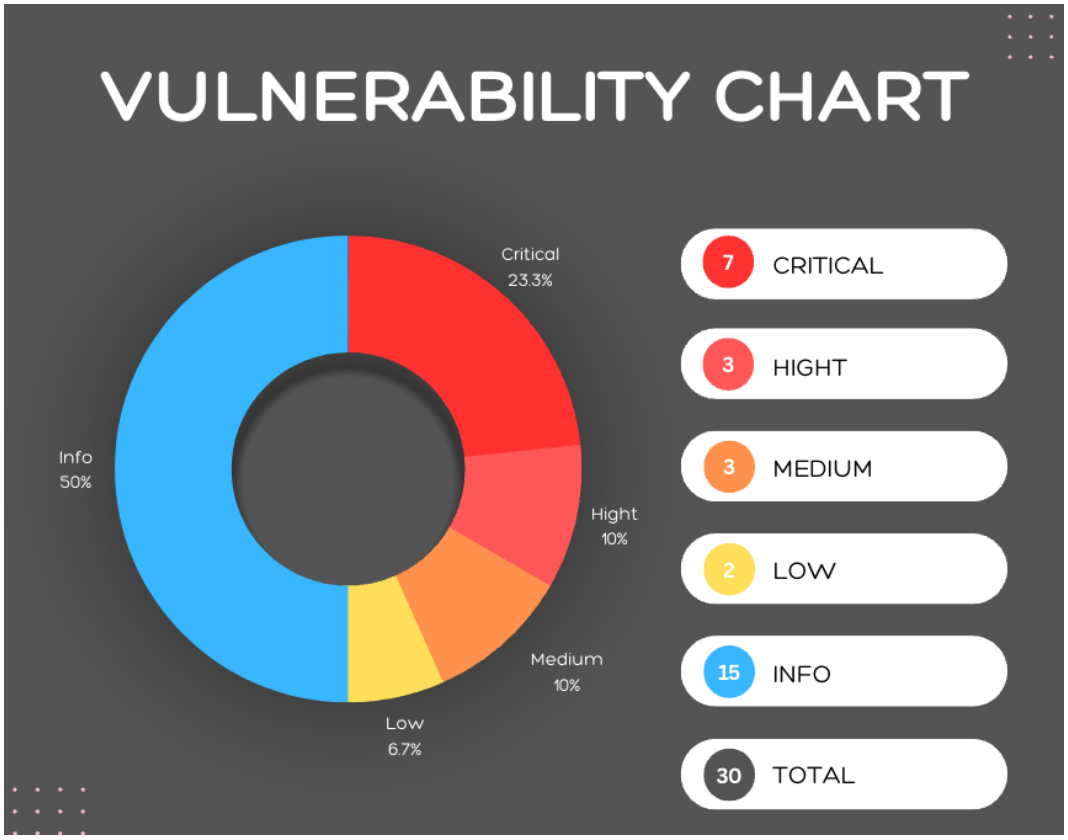

Figura 5.1: Grafico a torta delle vulnerabilità

Il grafico a torta consente di valutare in modo immediato la distribuzione delle vulnerabilità tra le diverse categorie, evidenziando la proporzione relativa di vulnerabilità critiche, elevate, medie, basse e informative. Questa visualizzazione fornisce un'ulteriore comprensione della distribuzione delle vulnerabilità, consentendo di identificare le aree che richiedono una maggiore attenzione e priorità nell'ambito delle attività di mitigazione e protezione del sistema.

CAPITOLO 6

Detailed Summary

Nella fase di Detailed Summary saranno presentate in elenco e descritte tutte le vulnerabilità identificate mediante l'utilizzo dei tool Nessus[1] e OWASP ZAP. Le vulnerabilità saranno categorizzate in base al livello di criticità che presentano. Sarà fornita una descrizione dettagliata di ciascuna vulnerabilità, includendo informazioni pertinenti come le cause sottostanti, le potenziali conseguenze e le raccomandazioni per la mitigazione. Questa analisi approfondita consentirà di comprendere appieno la portata delle vulnerabilità individuate e guidare le decisioni e le azioni necessarie per garantire la sicurezza e la protezione del sistema.

[1] https://github.com/vincenzo-esposit0/WEB_MACHINE_N7/tree/main/Report/VulnerabilityMapping/Nessus

6.1 Critical Vulnerability

6.1.1 Apache 2.4.x < 2.4.47 Multiple Vulnerabilities (Nessus)

Descrizione

La versione di Apache httpd installata sull'host remoto è precedente alla versione 2.4.47. Pertanto, è affetta da diverse vulnerabilità come indicate nel log delle modifiche della versione 2.4.47:

- Inatteso <Location> sezione di corrispondenza con 'MergeSlashes OFF'. (CVE-2021-30641 [1])
- mod_auth_digest: possibile overflow dello stack a causa di un byte nullo durante la convalida del nonce Digest. (CVE-2020-35452 [2])
- mod_session: crash a causa di dereferenziazione di un puntatore NULL, che potrebbe essere utilizzato per causare un Denial of Service con un server backend malevolo e SessionHeader. (CVE-2021-26691 [3])
- mod_session: crash a causa di dereferenziazione di un puntatore NULL, che potrebbe essere utilizzato per causare un Denial of Service. (CVE-2021-26690 [4])
- mod_proxy_http: crash a causa di dereferenziazione di un puntatore NULL, che potrebbe essere utilizzato per causare un Denial of Service. (CVE-2020-13950 [5])
- Windows: Prevenzione degli utenti locali dall'arresto del processo httpd (CVE-2020-13938).
- mod_proxy_wstunnel, mod_proxy_http: Gestione della negoziazione end-to-end dei protocolli aggiornabili. (CVE-2019-17567 [6])

Soluzione

Si consiglia di effettuare l'aggiornamento alla versione 2.4.47 o successiva di Apache.

6.1.2 Apache 2.4.x < 2.4.52 mod_lua Buffer Overflow (Nessus)

Descrizione

La versione di Apache httpd installata sull'host remoto è precedente alla versione 2.4.52. Pertanto, è affetta da una vulnerabilità correlata a mod_lua durante la gestione di contenuti multipart. Una richiesta con un corpo attentamente elaborato può causare un buffer overflow nel parser multipart di mod_lua (r:parsebody() chiamato da script Lua). Il team di Apache httpd non è a conoscenza di un exploit per la vulnerabilità, anche se potrebbe essere possibile crearne uno. (CVE-2021-44790 [7])

Soluzione

Si consiglia di effettuare l'aggiornamento alla versione 2.4.52 o successiva di Apache.

6.1.3 Apache 2.4.x >= 2.4.7 / < 2.4.52 Forward Proxy DoS / SSRF (Nessus)

Descrizione

La versione di Apache httpd installata sull'host remoto è uguale o superiore a 2.4.7 e precedente a 2.4.52.

Pertanto, è affetta da una falla legata all'agire come proxy inoltrato. Un URI manipolato inviato a httpd configurato come proxy inoltrato (ProxyRequests on) può causare un crash (dereferenziazione di puntatore NULL) o, per configurazioni che mescolano dichiarazioni di proxy inoltrato e proxy inverso, può consentire di indirizzare le richieste a un endpoint dichiarato di Unix Domain Socket (Server Side Request Forgery). (CVE-2021-44224)

Soluzione

Si consiglia di effettuare l'aggiornamento alla versione 2.4.52 o successiva di Apache.

6.1.4 Apache 2.4.x < 2.4.53 Multiple Vulnerabilities (Nessus)

Descrizione

La versione di Apache httpd installata sull'host remoto è precedente alla versione 2.4.53. Pertanto, è affetta da diverse vulnerabilità come indicate nell'advisory della versione 2.4.53:

- mod_lua: Utilizzo di un valore non inizializzato in r:parsebody: Una richiesta body opportunamente costruita può causare la lettura di un'area di memoria casuale, il che potrebbe causare il crash del processo. Questo problema interessa Apache HTTP Server 2.4.52 e versioni precedenti. (CVE-2022-22719 [8])

- HTTP request smuggling: Apache HTTP Server 2.4.52 e versioni precedenti non chiudono la connessione in ingresso quando si verificano errori eliminando il corpo della richiesta, esponendo il server a HTTP Request Smuggling. (CVE-2022-22720 [9])

- Possibile overflow del buffer con LimitXMLRequestBody molto grande o illimitato in core: Se LimitXMLRequestBody è impostato per consentire richieste con body più grandi di 350 MB (valore predefinito 1M) su sistemi a 32 bit, si verifica un overflow dell'intero che successivamente provoca scritture fuori dai limiti. Questo problema interessa Apache HTTP Server 2.4.52 e versioni precedenti. (CVE-2022-22721 [10])

- Lettura/scrittura al di là dei limiti in mod_sed: Vulnerabilità di scrittura al di fuori dei limiti in mod_sed di Apache HTTP Server consente a un attaccante di sovrascrivere la memoria heap con dati possibilmente forniti dall'attaccante. Questo problema interessa Apache HTTP Server 2.4 versione 2.4(.52 e versioni precedenti. (CVE-2022-23943 [11])

Soluzione

Si consiglia di effettuare l'aggiornamento alla versione 2.4.53 o successiva di Apache.

6.1.5 Apache 2.4.x < 2.4.54 Multiple Vulnerabilities (Nessus)

Descrizione

La versione di Apache httpd installata sull'host remoto è precedente alla versione 2.4.54. Pertanto, è affetta da diverse vulnerabilità come indicate nella nota informativa della versione 2.4.54.

- Possibile smistamento di richieste in mod_proxy_ajp: la vulnerabilità di interpretazione inconsistente delle richieste HTTP ("HTTP Request Smuggling") in mod_proxy_ajp di Apache HTTP Server consente a un attaccante di smistare richieste verso il server AJP a cui vengono inoltrate le richieste. Questo problema riguarda Apache HTTP Server versione 2.4.53 e versioni precedenti. (CVE-2022-26377 [12])

- Lettura oltre i limiti in mod_isapi: Apache HTTP Server 2.4.53 e versioni precedenti su Windows potrebbero leggere oltre i limiti quando configurati per elaborare richieste con il modulo mod_isapi. (CVE-2022-28330 [13])

- Lettura oltre i limiti tramite ap_rwrite(): la funzione ap_rwrite() in Apache HTTP Server 2.4.53 e versioni precedenti potrebbe leggere in modo imprevisto la memoria se un attaccante può indurre il server a riflettere un input molto grande usando ap_rwrite() o ap_rputs(), ad esempio con la funzione r:puts() di mod_lua. (CVE-2022-28614 [14])

- Lettura oltre i limiti in ap_strcmp_match(): Apache HTTP Server 2.4.53 e versioni precedenti potrebbero causare un arresto anomalo o una divulgazione di informazioni a causa di una lettura oltre i limiti in ap_strcmp_match() quando viene fornito un buffer di input estremamente grande. Sebbene nessun codice distribuito con il server possa essere indotto a effettuare una chiamata del genere, teoricamente potrebbero essere interessati moduli di terze parti o script lua che utilizzano ap_strcmp_match(). (CVE-2022-28615 [15])

- Denial of service in mod_lua r:parsebody: in Apache HTTP Server 2.4.53 e versioni precedenti, una richiesta malevola a uno script lua che chiama r:parsebody(0) potrebbe causare un denial of service a causa dell'assenza di un limite predefinito sulla dimensione possibile dell'input. (CVE-2022-29404 [16])

- Denial of Service mod_sed: se Apache HTTP Server 2.4.53 è configurato per effettuare trasformazioni con mod_sed in contesti in cui l'input per mod_sed potrebbe essere

molto grande, mod_sed potrebbe effettuare allocazioni di memoria eccessivamente grandi e provocare un arresto anomalo. (CVE-2022-30522 [17])

- Divulgazione di informazioni in mod_lua con websockets: Apache HTTP Server 2.4.53 e versioni precedenti potrebbero restituire lunghezze alle applicazioni che chiamano r:wsread() che puntano oltre la fine dello spazio allocato per il buffer. (CVE-2022-30556 [18])

- X-Forwarded-For omesso dal meccanismo hop-by-hop in mod_proxy: Apache HTTP Server 2.4.53 e versioni precedenti potrebbero non inviare gli header X-Forwarded-* al server di origine in base al meccanismo hop-by-hop dell'header di connessione lato client. Ciò potrebbe essere utilizzato per eludere l'autenticazione basata sull'IP sul server o sull'applicazione di origine. (CVE-2022-31813 [19])

Soluzione

Si consiglia di effettuare l'aggiornamento alla versione 2.4.54 o successiva di Apache.

6.1.6 Apache 2.4.x < 2.4.56 Multiple Vulnerabilities (Nessus)

Descrizione

La versione di Apache httpd installata sull'host remoto è precedente alla versione 2.4.56. Pertanto, è affetta da diverse vulnerabilità come indicate nella comunicazione relativa alla versione 2.4.56.

- Smuggling delle richieste HTTP con mod_rewrite e mod_proxy: Alcune configurazioni di mod_proxy su Apache HTTP Server nelle versioni da 2.4.0 a 2.4.55 consentono un attacco di Smuggling delle Richieste HTTP. Le configurazioni sono interessate quando mod_proxy è abilitato insieme a qualche forma di RewriteRule o ProxyPassMatch in cui un pattern non specifico corrisponde a una parte dei dati della richiesta-target (URL) forniti dall'utente e viene quindi reinserito nella richiesta-target del proxy utilizzando la sostituzione delle variabili. (CVE-2023-25690 [20])

- Apache HTTP Server: Smuggling delle risposte HTTP in mod_proxy_uwsgi: Vulnerabilità di Smuggling delle Risposte HTTP in Apache HTTP Server tramite mod_proxy_uwsgi. Questo problema interessa Apache HTTP Server: dalla versione 2.4.30 alla 2.4.55. I caratteri speciali nell'intestazione di risposta di origine possono troncare/dividere la risposta inoltrata al client. (CVE-2023-27522 [21])

Soluzione

Si consiglia di effettuare l'aggiornamento alla versione 2.4.56 o successiva di Apache.

6.1.7 Apache < 2.4.49 Multiple Vulnerabilities (Nessus)

Descrizione

La versione di Apache httpd installata sull'host remoto è precedente alla versione 2.4.49. Pertanto, è affetta da una vulnerabilità come indicato nel log delle modifiche della versione 2.4.49:

- Una richiesta uri-path manipolata può causare a mod_proxy l'inoltro della richiesta a un server di origine scelto dall'utente remoto. (CVE-2021-40438 [22])
- La funzione ap_escape_quotes() potrebbe scrivere oltre la fine di un buffer quando viene fornito un input malevolo. Nessun modulo incluso passa dati non attendibili a queste funzioni, ma moduli di terze parti/esterni potrebbero farlo. (CVE-2021-39275 [23])
- Richieste malformate potrebbero causare il dereferenziamento di un puntatore NULL da parte del server. (CVE-2021-34798 [24])

Soluzione

Si consiglia di effettuare l'aggiornamento alla versione 2.4.49 o successiva di Apache.

6.2 Hight Vulnerability

6.2.1 Apache >= 2.4.17 < 2.4.49 mod_http2 (Nessus)

Descrizione

La versione di Apache httpd installata sull'host remoto è superiore a 2.4.17 e precedente a 2.4.49. Pertanto, è affetta da una vulnerabilità come indicato nel log delle modifiche della versione 2.4.49. Un metodo manipolato inviato tramite HTTP/2 bypasserà la convalida e verrà inoltrato da mod_proxy, il che può portare alla divisione delle richieste o all'inquinamento della cache. (CVE-2021-36160 [25])

Soluzione

Si consiglia di effettuare l'aggiornamento alla versione 2.4.49 o successiva di Apache.

6.2.2 Apache >= 2.4.30 < 2.4.49 mod_proxy_uwsgi (Nessus)

Descrizione

La versione di Apache httpd installata sull'host remoto è superiore a 2.4.30 e precedente a 2.4.49. Pertanto, è affetta da una vulnerabilità come indicato nel log delle modifiche della versione 2.4.49. Una richiesta URI-path attentamente creata può causare a mod_proxy_uwsgi la lettura di una quantità di memoria superiore a quella allocata e provocare un arresto anomalo (DoS). (CVE-2021-33193 [26])

Soluzione

Si consiglia di effettuare l'aggiornamento alla versione 2.4.49 o successiva di Apache.

6.2.3 SQL Injection - MySQL (OWSAP-ZAP)

Descrizione

Non fidarsi dell'input del client, anche se è presente una validazione lato client. In generale, verificare il tipo di tutti i dati lato server.

- Se l'applicazione utilizza JDBC, utilizzare PreparedStatement o CallableStatement, con parametri passati tramite '?'.
- Se l'applicazione utilizza ASP, utilizzare oggetti di comando ADO con un forte controllo dei tipi e query parametrizzate.
- Se è possibile utilizzare Stored Procedure del database.
- Non concatenare le stringhe nelle query nella stored procedure o utilizzare 'exec', 'exec immediate' o funzionalità equivalenti.
- Esegui l'escape di tutti i dati ricevuti dal client.

Soluzione

Applicare una "lista di caratteri ammessi" o una "lista di caratteri negati" non ammessi nell'input dell'utente. Applicare il principio del privilegio minimo utilizzando l'utente del database con i privilegi minimi possibili.

Evitare di utilizzare gli utenti del database 'sa' o 'db-owner'. Ciò non elimina l'iniezione SQL, ma ne riduce al minimo l'impatto. Concedere il minimo accesso al database necessario per l'applicazione. (CVE-2019-11510 [27], CVE-2019-16759 [28], CVE-2020-11651 [29], CVE-2021-29478 [30], CVE-2022-27927 [31])

6.3 Medium Vulnerability

6.3.1 Assenza di Token Anti-CSRF (OWASP-ZAP)

Nessun Token Anti-CSRF è stato rilevato nel form HTML.

Descrizione

Un attacco di cross-site request forgery (CSRF) è un tipo di attacco che consiste nel costringere una vittima a inviare una richiesta HTTP a una destinazione di destinazione senza il loro consenso o intento al fine di eseguire un'azione come se fosse la vittima stessa. La causa sottostante è la funzionalità dell'applicazione che utilizza URL/azioni di form prevedibili in modo ripetitivo. La natura dell'attacco è che il CSRF sfrutta la fiducia che un sito web ha per un utente. Al contrario, il cross-site scripting (XSS) sfrutta la fiducia che un utente ha per un sito web. Come l'XSS, gli attacchi CSRF non sono necessariamente cross-site, ma possono esserlo. Il cross-site request forgery è anche conosciuto come CSRF, XSRF, attacco a un clic, session riding, vice confuso e sea surf.

Gli attacchi CSRF sono efficaci in diverse situazioni, tra cui:

- La vittima ha una sessione attiva sul sito di destinazione.
- La vittima è autenticata tramite HTTP auth sul sito di destinazione.
- La vittima si trova nella stessa rete locale del sito di destinazione.

Il CSRF è stato principalmente utilizzato per eseguire un'azione contro un sito di destinazione utilizzando i privilegi della vittima, ma di recente sono state scoperte tecniche per divulgare informazioni ottenendo accesso alla risposta. Il rischio di divulgazione delle informazioni aumenta notevolmente quando il sito di destinazione è vulnerabile all'XSS, poiché l'XSS può essere utilizzato come piattaforma per il CSRF, consentendo all'attacco di operare all'interno dei limiti della stessa politica di origine. (CVE-2022-32555 [32])

Soluzione

Migliorare la fase di architettura e design del sistema sviluppato.

6.3.2 Content Security Policy (CSP) Header Not Set (OWASP-ZAP)

Descrizione

La Content Security Policy (CSP) è un ulteriore livello di sicurezza che aiuta a rilevare e mitigare determinati tipi di attacchi, tra cui Cross Site Scripting (XSS) e attacchi di iniezione di dati. Questi attacchi vengono utilizzati per rubare dati, vandalizzare siti o distribuire malware. La CSP fornisce un insieme di intestazioni HTTP standard che consentono ai proprietari dei siti web di dichiarare le origini approvate dei contenuti che i browser devono essere autorizzati a caricare sulla pagina, tra cui JavaScript, CSS, frame HTML, font, immagini e oggetti incorporabili come applet Java, ActiveX, file audio e video.

Soluzione

Assicurarsi che il server web, il server dell'applicazione, il bilanciatore di carico, ecc. siano configurati per impostare l'intestazione Content-Security-Policy.

6.3.3 Missing Anti-clickjacking Header (OWASP-ZAP)

Descrizione

La risposta non include né Content-Security-Policy con la direttiva 'frame-ancestors', né X-Frame-Options per proteggersi dagli attacchi di 'ClickJacking'. (CVE-2023-3140 [33])

Soluzione

I moderni browser web supportano gli header HTTP Content-Security-Policy e X-Frame-Options. Assicurarsi che uno di essi sia impostato su tutte le pagine web restituite dal tuo sito/app.

6.4 Low Vulnerability

6.4.1 Server Leaks Version Information via "Server" HTTP Response Header Field (OWASP-ZAP)

Descrizione

Il server web/applicazione sta rivelando informazioni sulla versione tramite l'intestazione di risposta HTTP "Server". L'accesso a tali informazioni potrebbe facilitare agli attaccanti l'identificazione di altre vulnerabilità a cui il tuo server web/applicazione è soggetto. (CVE-2014-0094 [34])

Soluzione

Assicurarsi che il tuo server web, server di applicazione, bilanciatore di carico, ecc. sia configurato per sopprimere l'intestazione "Server" o fornire dettagli generici.

6.4.2 X-Content-Type-Options Header Missing (OWASP-ZAP)

Descrizione

L'intestazione Anti-MIME-Sniffing X-Content-Type-Options non è stata impostata su 'nosniff'. Ciò consente alle versioni precedenti di Internet Explorer e Chrome di eseguire il MIME-sniffing sul corpo della risposta, potenzialmente causando l'interpretazione e la visualizzazione del corpo della risposta come un tipo di contenuto diverso dal tipo di contenuto dichiarato. Le versioni attuali (primi del 2014) e le versioni legacy di Firefox utilizzeranno il tipo di contenuto dichiarato (se impostato), anziché eseguire il MIME-sniffing.

Soluzione

Assicurarsi che l'applicazione/server web imposti correttamente l'intestazione Content-Type e che imposti l'intestazione X-Content-Type-Options su 'nosniff' per tutte le pagine web. Se possibile, assicurarsi che l'utente finale utilizzi un browser web compatibile con gli standard e moderno che non esegue affatto il MIME-sniffing, o che possa essere direzionato dall'applicazione web/server web per non eseguire il MIME-sniffing.

6.5 Info Vulnerability

6.5.1 GET for POST (OWASP-ZAP)

Descrizione

È stata osservata una richiesta che inizialmente era di tipo POST ma è stata accettata anche come GET. Questo problema non rappresenta di per sé una vulnerabilità di sicurezza, tuttavia potrebbe agevolare la semplificazione di altri attacchi. Ad esempio, se il POST originale è soggetto a Cross-Site Scripting (XSS), allora questa scoperta potrebbe indicare la possibilità di un attacco semplificato (basato su GET) di XSS.

6.5.2 User Controllable HTML Element Attribute (Potential XSS) (OWASP-ZAP)

Descrizione

Questa verifica analizza l'input fornito dall'utente nei parametri della stringa di query e nei dati inviati tramite POST per identificare i punti in cui determinati valori degli attributi HTML potrebbero essere controllati. Ciò fornisce una rilevazione dei punti critici per gli attacchi XSS (cross-site scripting) che richiederà una revisione più approfondita da parte di un analista di sicurezza per determinarne l'eventuale sfruttabilità.

6.5.3 HTTP Server Type and Version (Nessus)

Descrizione

Questo plugin cerca di determinare il tipo e la versione del server web remoto.

6.5.4 ICMP Timestamp Request Remote Date Disclosure (Nessus)

Descrizione

L'host remoto risponde a una richiesta di timestamp ICMP. Ciò consente a un attaccante di conoscere la data impostata sulla macchina di destinazione, il che potrebbe aiutare un attaccante remoto non autenticato a superare i protocolli di autenticazione basati sul tempo.

6.5.5 Traceroute Information (Nessus)

Descrizione

Effettua una traccia di percorso (traceroute) verso l'host remoto.

6.5.6 OS Identification (Nessus)

Descrizione

Attraverso l'uso di una combinazione di sonde remote (ad esempio, TCP/IP, SMB, HTTP, NTP, SNMP, ecc.), è possibile indovinare il nome del sistema operativo remoto in uso. In alcuni casi, è anche possibile indovinare la versione del sistema operativo.

6.5.7 Service Detection (Nessus)

Descrizione

Nessus è stato in grado di identificare il servizio remoto attraverso il suo banner o osservando il messaggio di errore che invia quando riceve una richiesta HTTP.

6.5.8 HyperText Transfer Protocol (HTTP) Information (Nessus)

Descrizione

Questo test fornisce alcune informazioni sul protocollo HTTP remoto: la versione utilizzata, se sono abilitate le funzionalità di HTTP Keep-Alive e HTTP pipelining, ecc...

6.5.9 TCP/IP Timestamps Supported (Nessus)

Descrizione

L'host remoto implementa i timestamp TCP, come definito da RFC1323. Un effetto collaterale di questa funzionalità è che talvolta è possibile calcolare l'uptime dell'host remoto.

6.5.10 HTTP Methods Allowed (per directory) (Nessus)

Descrizione

Chiamando il metodo OPTIONS, è possibile determinare quali metodi HTTP sono consentiti in ogni directory. I seguenti metodi HTTP sono considerati insicuri: PUT, DELETE, CONNECT, TRACE, HEAD. Molti framework e linguaggi trattano 'HEAD' come una richiesta 'GET', sebbene senza alcun corpo nella risposta. Se fosse stata impostata una restrizione di sicurezza sulle richieste 'GET' in modo che solo gli 'authenticatedUsers' potessero accedere alle richieste 'GET' per un particolare servlet o risorsa, questa restrizione verrebbe bypassata per la versione 'HEAD'. Ciò consentiva la sottomissione non autorizzata di qualsiasi richiesta privilegiata di tipo GET. Poiché questo elenco potrebbe essere incompleto, il plugin effettua

anche test - se sono abilitati i 'Thorough tests' o se 'Enable web applications tests' è impostato su 'yes' nella politica di scansione - utilizzando vari metodi HTTP noti su ciascuna directory e li considera non supportati se riceve un codice di risposta 400, 403, 405 o 501.

6.5.11 Common Platform Enumeration (CPE) (Nessus)

Descrizione

Utilizzando le informazioni ottenute da una scansione Nessus, il presente plugin segnala le corrispondenze CPE (Common Platform Enumeration) per vari prodotti hardware e software trovati su un host.

6.5.12 Apache HTTP Server Version (Nessus)

Descrizione

L'host remoto sta eseguendo Apache HTTP Server, un server web open source. È stato possibile leggere il numero di versione dal banner.

6.5.13 Device Type (Nessus)

Descrizione

In base al sistema operativo remoto, è possibile determinare quale sia il tipo di sistema remoto.

6.5.14 Patch Report (Nessus)

Descrizione

L'host remoto presenta una o più patch di sicurezza mancanti. Questo plugin elenca la versione più recente di ciascuna patch da installare per garantire che l'host remoto sia aggiornato.

6.5.15 Ethernet MAC Addresses (Nessus)

Descrizione

Questo plugin raccoglie gli indirizzi MAC scoperti sia tramite il sondaggio remoto dell'host (ad esempio, SNMP e Netbios) sia dall'esecuzione di controlli locali (ad esempio, ifconfig). Successivamente, consolida gli indirizzi MAC in un'unica lista univoca e uniforme.

Bibliografia

[1] CVE Details, "Vulnerability details : CVE-2021-30641." https://nvd.nist.gov/vuln/detail/CVE-2021-30641. (Citato a pagina 11)

[2] CVE Details, "Vulnerability details : CVE-2020-35452." https://nvd.nist.gov/vuln/detail/CVE-2020-35452. (Citato a pagina 11)

[3] CVE Details, "Vulnerability details : CVE-2021-26691." https://nvd.nist.gov/vuln/detail/CVE-2021-26691. (Citato a pagina 11)

[4] CVE Details, "Vulnerability details : CVE-2021-26690." https://nvd.nist.gov/vuln/detail/CVE-2021-26690. (Citato a pagina 11)

[5] CVE Details, "Vulnerability details : CVE-2020-13950." https://nvd.nist.gov/vuln/detail/CVE-2020-13950. (Citato a pagina 11)

[6] CVE Details, "Vulnerability details : CVE-2019-17567." https://nvd.nist.gov/vuln/detail/CVE-2019-17567. (Citato a pagina 11)

[7] CVE Details, "Vulnerability details : CVE-2021-44790." https://nvd.nist.gov/vuln/detail/CVE-2021-44790. (Citato a pagina 12)

[8] CVE Details, "Vulnerability details : CVE-2022-22719." https://nvd.nist.gov/vuln/detail/CVE-2022-22719. (Citato a pagina 13)

[9] CVE Details, "Vulnerability details : CVE-2022-22720." https://nvd.nist.gov/vuln/detail/CVE-2022-22720. (Citato a pagina 13)

[10] CVE Details, "Vulnerability details : CVE-2022-22721." https://nvd.nist.gov/vuln/detail/CVE-2022-22721. (Citato a pagina 13)

[11] CVE Details, "Vulnerability details : CVE-2022-23943." https://nvd.nist.gov/vuln/detail/CVE-2022-23943. (Citato a pagina 13)

[12] CVE Details, "Vulnerability details : CVE-2022-26377." https://nvd.nist.gov/vuln/detail/CVE-2022-26377. (Citato a pagina 14)

[13] CVE Details, "Vulnerability details : CVE-2022-28330." https://nvd.nist.gov/vuln/detail/CVE-2022-28330. (Citato a pagina 14)

[14] CVE Details, "Vulnerability details : CVE-2022-28614." https://nvd.nist.gov/vuln/detail/CVE-2022-28614. (Citato a pagina 14)

[15] CVE Details, "Vulnerability details : CVE-2022-28615." https://nvd.nist.gov/vuln/detail/CVE-2022-28615. (Citato a pagina 14)

[16] CVE Details, "Vulnerability details : CVE-2022-29404." https://nvd.nist.gov/vuln/detail/CVE-2022-29404. (Citato a pagina 14)

[17] CVE Details, "Vulnerability details : CVE-2022-30522." https://nvd.nist.gov/vuln/detail/CVE-2022-30522. (Citato a pagina 15)

[18] CVE Details, "Vulnerability details : CVE-2022-30556." https://nvd.nist.gov/vuln/detail/CVE-2022-30556. (Citato a pagina 15)

[19] CVE Details, "Vulnerability details : CVE-2022-31813." https://nvd.nist.gov/vuln/detail/CVE-2022-31813. (Citato a pagina 15)

[20] CVE Details, "Vulnerability details : CVE-2023-25690." https://nvd.nist.gov/vuln/detail/CVE-2023-25690. (Citato a pagina 16)

[21] CVE Details, "Vulnerability details : CVE-2023-27522." https://nvd.nist.gov/vuln/detail/CVE-2023-27522. (Citato a pagina 16)

[22] CVE Details, "Vulnerability details : CVE-2021-40438." https://nvd.nist.gov/vuln/detail/CVE-2021-40438. (Citato a pagina 17)

[23] CVE Details, "Vulnerability details : CVE-2021-39275." https://nvd.nist.gov/vuln/detail/CVE-2021-39275. (Citato a pagina 17)

[24] CVE Details, "Vulnerability details : CVE-2021-34798." https://nvd.nist.gov/vuln/detail/CVE-2021-34798. (Citato a pagina 17)

[25] CVE Details, "Vulnerability details : CVE-2021-36160." https://nvd.nist.gov/vuln/detail/CVE-2021-36160. (Citato a pagina 18)

[26] CVE Details, "Vulnerability details : CVE-2021-33193." https://nvd.nist.gov/vuln/detail/CVE-2021-33193. (Citato a pagina 18)

[27] CVE Details, "Vulnerability details : CVE-2019-11510." https://nvd.nist.gov/vuln/detail/CVE-2019-11510. (Citato a pagina 19)

[28] CVE Details, "Vulnerability details : CVE-2019-16759." https://nvd.nist.gov/vuln/detail/CVE-2019-16759. (Citato a pagina 19)

[29] CVE Details, "Vulnerability details : CVE-2020-11651." https://nvd.nist.gov/vuln/detail/CVE-2020-11651. (Citato a pagina 19)

[30] CVE Details, "Vulnerability details : CVE-2021-29478." https://nvd.nist.gov/vuln/detail/CVE-2021-29478. (Citato a pagina 19)

[31] CVE Details, "Vulnerability details : CVE-2022-27927." https://nvd.nist.gov/vuln/detail/CVE-2022-27927. (Citato a pagina 19)

[32] CVE Details, "Vulnerability details : CVE-2022-32555." https://nvd.nist.gov/vuln/detail/CVE-2022-32555. (Citato a pagina 20)

[33] CVE Details, "Vulnerability details : CVE-2023-3140." https://nvd.nist.gov/vuln/detail/CVE-2023-3140. (Citato a pagina 21)

[34] CVE Details, "Vulnerability details : CVE-2014-0094." https://nvd.nist.gov/vuln/detail/CVE-2014-0094. (Citato a pagina 22)