Proof Of Concept

Linux Security- Exploitation & Hardening

Task 6: Log Analysis & Intrusion Detection

1. Executive Summary

This PoC demonstrates how to analyze system logs to detect and mitigate brute-force SSH login attempts. The task involves enabling system logging, simulating failed login attempts, analyzing logs for intrusion detection, and implementing fail2ban to block repeated failed attempts.

2. Objectives

- *Setup*: Enable system logging and simulate multiple failed SSH login attempts.
- *Exploit*: Analyze logs to identify brute-force attempts and unauthorized access.
- *Mitigation*: Implement fail2ban to block repeated failed attempts and set up log monitoring automation.

3. Setup

3.1 Enable System Logging

System logging was enabled using rsyslog to capture authentication logs in /var/log/auth.log.

Commands Used:

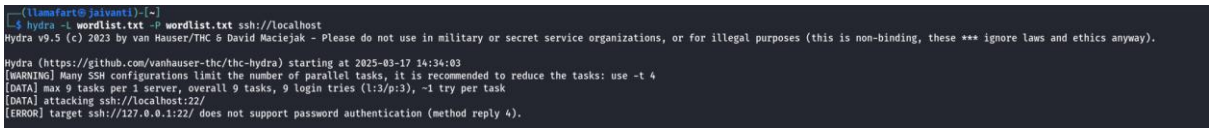
sudo systemctl status rsyslog

3.2 Simulate Failed SSH Login Attempts

Multiple failed SSH login attempts were simulated to generate log entries for analysis.

Commands Used:

hydra -l invaliduser -P world1k.txt ssh://localhost



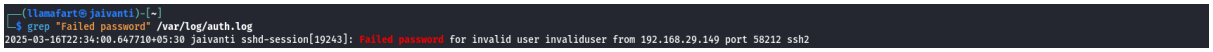
4. Exploitation

4.1 Analyze Logs for Failed Login Attempts

The /var/log/auth.log file was analyzed to identify failed SSH login attempts.

Commands Used:

grep "Failed password" /var/log/auth.log



5. Mitigation

5.1 Implement fail2ban

fail2ban was configured to block IP addresses with repeated failed login attempts.

Commands Used:

sudo nano /etc/fail2ban/jail.local

sudo systemctl restart fail2ban

5.2 Set Up Log Monitoring Automation

Log monitoring was automated using logwatch and rsyslog to ensure continuous monitoring and alerting.

Commands Used:

sudo logwatch --detail high --mailto user@example.com --range today

sudo systemctl status rsyslog

6. Conclusion

This PoC successfully demonstrated how to detect brute-force SSH login attempts through log analysis and mitigate the risk using fail2ban. By automating log monitoring, the system can proactively respond to potential security threats.