Blocking Brute Force Attacks with fail2ban

LPIC-2: Linux Engineer (202-450)

Objectives:

At the end of this episode, I will be able to:

1. Install and configure fail2ban to limit login attempts.
2. Test fail2ban using a brute force attack.

> Additional resources used during the episode can be obtained using the download link on the overview episode.

- Blocking Brute Force Attacks with *fail2ban*
    - What is *fail2ban*
    - Installing and configuring *fail2ban*
    - Testing and monitoring *fail2ban*
- *fail2ban*
    - Daemon that runs in the background
    - Examines log files for bad activity
    - Bans IPs by interacting with the firewall
        - Supports iptables and firewalld
    - Used a lot with FreePBX and SIP services
- Installing *fail2ban*
    1. Install fail2ban
        - `sudo apt install fail2ban`
        - Depends on
            - `python3`
    2. Enable and start the daemon
        - `sudo systemctl enable --now fail2ban`
- Auditing logins
    1. Determine which services we want to protect
        - Full list of services supported is in `/etc/fail2ban/filter.d`
    2. Examine the default config for documentation
        - `/etc/fail2ban/jail.conf`
    3. Create a customized config for our desired services
        - `/etc/fail2ban/jail.local`
        - Overrides jail.conf
- Configure fail2ban for SSH
    - `sudoedit /etc/fail2ban/jail.local`
        - `[DEFAULT]`
        - `bantime = 1h`
        - `banaction = ufw`
        - `[sshd]`
        - `enabled = true`
    - Two typical actions
        - Ban the IP
            - `%(action_)s`

        - Ban the IP and send an email notification w/logs
            - `%(action_mwl)s`
        - More actions in `/etc/fail2ban/jail.conf`
- Monitoring fail2ban
    - Examine the fail2ban log files
        - `sudo tail /var/log/fail2ban.log`
    - Query the status with the fail2ban client utility
        - `sudo fail2ban-client status`
        - `sudo fail2ban-client status sshd`
    - Examine the fail2ban entried in journald
        - `sudo journalctl -xau fail2ban`
- Performing a test attack
    1. Create a test user and start monitoring
        - `sudo adduser jdoe`
            - User password "thunder"
        - `sudo tail /var/log/fail2ban.log`
    2. Perform a login attack from another host
        - 10,000 most common passwords file (https://raw.githubusercontent.com/danielmiessler/SecLists/master/Passwords/Common-Credentials/10k-most-common.txt)
        - `hydra -l dpezet -t 2 -P ./passlist.txt 172.16.0.237 ssh`
            - `-l` User to attack
            - `-t` Number of simultaneous attempts
            - `-p` File containing a list of passwords
                - `-x` Allows dynamically generating passwords
            - `<ip>`
            - `<service>`
    3. View logs (step 1) to see the failed login messages
    4. Verify the ban is in place
        - `sudo fail2ban-client status sshd`
- Whitelisting an IP
    - `vi /etc/fail2ban/jail.local`
    - `ignoreip = 127.0.0.1/8`