

TASK1: LAB Assignment

LAB 1: Linux Security Configuration

Task 1: สร้าง User Accounts สำหรับ Team (นายนักพัฒนา สมบัติวงศ์)

1.1 สร้าง Users และ Groups:

สร้าง groups

sudo groupadd developers

sudo groupadd testers

sudo groupadd dbadmin

```
dev@LinuxServer:~$ sudo groupadd developers
sudo groupadd testers
sudo groupadd dbadmin
[sudo] password for dev:
Sorry, try again.
[sudo] password for dev:
dev@LinuxServer:~$
```

สร้าง users

sudo useradd -m -s /bin/bash -G developers chonl

sudo useradd -m -s /bin/bash -G developers nunt

sudo useradd -m -s /bin/bash -G testers tuser

sudo useradd -m -s /bin/bash -G dbadmin dbuser

```
dev@LinuxServer:~$ sudo useradd -m -s /bin/bash -G developers chonl
sudo useradd -m -s /bin/bash -G developers nunt
sudo useradd -m -s /bin/bash -G testers tuser
sudo useradd -m -s /bin/bash -G dbadmin dbuser
dev@LinuxServer:~$
```

ตั้งรหัสผ่าน (ต้องตาม policy)

sudo passwd chonl

sudo passwd nunt

sudo passwd tuser

sudo passwd dbuser

```
dev@LinuxServer:~$ sudo passwd chonl
New password:
Retype new password:
passwd: password updated successfully
dev@LinuxServer:~$ sudo passwd nunt
New password:
Retype new password:
passwd: password updated successfully
dev@LinuxServer:~$ sudo passwd tuser
New password:
Retype new password:
passwd: password updated successfully
dev@LinuxServer:~$ sudo passwd dbuser
New password:
Retype new password:
passwd: password updated successfully
dev@LinuxServer:~$
```

1.2 ตั้งค่า Password Policy:

แก้ไขไฟล์ /etc/login.defs
sudo nano /etc/login.defs

เปลี่ยนค่าเหล่านี้:

```
PASS_MAX_DAYS 90
PASS_MIN_DAYS 7
PASS_WARN_AGE 14
PASS_MIN_LEN 12
```

```
#
# Password aging controls:
#
#     PASS_MAX_DAYS   Maximum number of days a password may be used.
#     PASS_MIN_DAYS   Minimum number of days allowed between password changes.
#     PASS_WARN_AGE   Number of days warning given before a password expires.
#
PASS_MAX_DAYS 90
PASS_MIN_DAYS 7
PASS_WARN_AGE 14
PASS_MIN_LEN 12
```

ติดตั้ง libpam-pwquality

```
sudo apt install libpam-pwquality
```

```
dev@LinuxServer:~$ sudo apt install libpam-pwquality
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  cracklib-runtime libcrack2 libpwquality-common libpwquality1 wamerican
The following NEW packages will be installed:
  cracklib-runtime libcrack2 libpam-pwquality libpwquality-common libpwquality1 wamerican
0 upgraded, 6 newly installed, 0 to remove and 3 not upgraded.
Need to get 446 kB of archives.
After this operation, 1,932 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

```
# แก้ไข /etc/pam.d/common-password
```

```
sudo nano /etc/pam.d/common-password
```

```
# เพิ่มบรรทัด:
```

```
password requisite pam_pwquality.so retry=3 minlen=12 difok=3 ucredit=-1 lcredit=-1
```

```
dccredit=-1 ocredit=-1
```

```
# here are the per-package modules (the "Primary" block)
password      requisite          pam_pwquality.so retry=3 minlen=12 difok=3 ucredit=-1 lcredit=-1 dcredit=-1
password      [success=1 default=ignore] pam_unix.so obscure use_authtok try_first_pass yescrypt
# here's the fallback if no module succeeds
password      requisite          pam_deny.so
# prime the stack with a positive return value if there isn't one already;
# this avoids us returning an error just because nothing sets a success code
# since the modules above will each just jump around
```

```
cat /etc/passwd | tail -4
```

```
groups chonl nunt tuser dbuser
```

```
dev@LinuxServer:~$ sudo nano /etc/pam.d/common-password
dev@LinuxServer:~$ cat /etc/passwd | tail -4
chonl:x:1001:1004::/home/chonl:/bin/bash
nunt:x:1002:1005::/home/nunt:/bin/bash
tuser:x:1003:1006::/home/tuser:/bin/bash
dbuser:x:1004:1007::/home/dbuser:/bin/bash
dev@LinuxServer:~$ groups chonl nunt tuser dbuser
chonl : chonl developers
nunt  : nunt developers
tuser : tuser testers
dbuser : dbuser dbadmin
```

Task 2: ตั้งค่า Sudo Permissions (นายณัฏฐพัฒน์ สมบัติวงศ์)

2.1 สร้าง Sudo Groups:

```
# สร้าง custom sudo groups
```

```
sudo groupadd sudo-developers
```

```
sudo groupadd sudo-limited
```

```
dev@LinuxServer:~$ sudo groupadd sudo-developers  
sudo groupadd sudo-limited
```

```
# เพิ่ม users เข้า groups
```

```
sudo usermod -aG sudo-developers chonl
```

```
sudo usermod -aG sudo-developers nunt
```

```
sudo usermod -aG sudo-limited tuser
```

```
dev@LinuxServer:~$ sudo usermod -aG sudo-developers chonl  
sudo usermod -aG sudo-developers nunt  
sudo usermod -aG sudo-limited tuser
```

2.2 Configure Sudoers:

```
# แก้ไขไฟล์ sudoers
```

```
sudo visudo
```

```
# เพิ่มกฎเหล่านี้:
```

```
# Developers - full sudo access
```

```
%sudo-developers ALL=(ALL:ALL) ALL
```

```
# Limited sudo - specific commands only
```

```
%sudo-limited ALL=(ALL) /usr/bin/systemctl status *, /usr/bin/tail /var/log/*, /bin/ps
```

```
# Database admin - database commands only
```

```
david ALL=(ALL) /usr/bin/mysql, /usr/bin/mysqldump, /bin/systemctl restart mysql
```

Sudo session timeout (15 minutes)

Defaults timestamp_timeout=15

Log sudo commands

Defaults logfile="/var/log/sudo.log"

Defaults log_input, log_output

```
# User privilege specification
root    ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin  ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo  ALL=(ALL:ALL) ALL

# Developers - full sudo access
%sudo-developers ALL=(ALL:ALL) ALL

# Limited sudo - specific commands only
%sudo-limited ALL=(ALL) /usr/bin/systemctl status *, /usr/bin/tail /var/log/*, /bin/ps

# Database admin - database commands only
david ALL=(ALL) /usr/bin/mysql, /usr/bin/mysqldump, /bin/systemctl restart mysql

# Sudo session timeout (15 minutes)
Defaults timestamp_timeout=15

# Log sudo commands
Defaults logfile="/var/log/sudo.log"
Defaults log_input, log_output
```

2.3 ทดสอบ Sudo Permissions:

ทดสอบด้วย chonl

sudo -u chonl sudo ls /root

```
dev@LinuxServer:~$ sudo -u chonl sudo ls /root
[sudo] password for chonl:
vboxpostinstall.sh
```

ทดสอบด้วย tuser (ควรใช้ได้เฉพาะคำสั่งที่อนุญาต)

sudo -u tuser sudo systemctl status ssh

```

dev@LinuxServer:~$ sudo -u tuser sudo systemctl status ssh
[sudo] password for tuser:
• ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enabled)
   Active: active (running) since Wed 2025-08-27 09:00:02 UTC; 29min ago
   TriggeredBy: • ssh.socket
   Docs: man:sshd(8)
         man:sshd_config(5)
   Process: 2162 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 2164 (sshd)
   Tasks: 1 (limit: 4605)
   Memory: 3.0M (peak: 4.0M)
   CPU: 179ms
   CGroup: /system.slice/ssh.service
           └─2164 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Aug 27 09:00:02 LinuxServer systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Aug 27 09:00:02 LinuxServer sshd[2164]: Server listening on 0.0.0.0 port 22.
Aug 27 09:00:02 LinuxServer sshd[2164]: Server listening on :: port 22.
Aug 27 09:00:02 LinuxServer systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
Aug 27 09:00:22 LinuxServer sshd[2166]: Received disconnect from 192.168.56.1 port 59305:11: Normal Shutdown [preauth]
Aug 27 09:00:22 LinuxServer sshd[2166]: Disconnected from 192.168.56.1 port 59305 [preauth]
Aug 27 09:02:33 LinuxServer sshd[2193]: Accepted password for dev from 192.168.56.1 port 59316 ssh2
Aug 27 09:02:33 LinuxServer sshd[2193]: pam_unix(sshd:session): session opened for user dev(uid=1000) by dev(uid=0)

```

sudo -u tuser sudo apt update # ควร fail

```

dev@LinuxServer:~$ sudo -u tuser sudo apt update
[sudo] password for tuser:
Sorry, user tuser is not allowed to execute '/usr/bin/apt update' as root on LinuxServer.

```

Task 3: Configure SSH Security (นายชลศักดิ์ กันทา)

3.1 Backup และแก้ไข SSH Config:

```

# Backup original config
sudo cp /etc/ssh/sshd_config /etc/ssh/sshd_config.backup

dev@LinuxServer:~$ sudo cp /etc/ssh/sshd_config /etc/ssh/sshd_config.backup
dev@LinuxServer:~$

```

แก้ไข SSH configuration

sudo nano /etc/ssh/sshd_config

เปลี่ยนค่าเหล่านี้:

Port 2222	# เปลี่ยนจาก default port
PermitRootLogin no	# ห้าม root login
PasswordAuthentication yes	# อนุญาต password (ชั่วคราว)
PubkeyAuthentication yes	# เปิดใช้ key-based auth
MaxAuthTries 3	# จำกัดความพยายาม

```
ClientAliveInterval 300      # Timeout session
ClientAliveCountMax 2        # Max idle sessions
AllowUsers chonl nunt tuser dbuser # อนุญาตเฉพาะ users เหล่านี้
Protocol 2                   # ใช้ SSH Protocol 2
```

```
#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::
# เปลี่ยน port จากค่า default (22 → 2222)
Port 2222

# ไม่อนุญาตให้ root login โดยตรง
PermitRootLogin no

# Authentication settings
PasswordAuthentication yes
PubkeyAuthentication yes
MaxAuthTries 3

# Idle session control
ClientAliveInterval 300
ClientAliveCountMax 2

# Allow specific users only
AllowUsers chonl nunt tuser dbuser

# Force SSH Protocol 2
Protocol 2

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key
```

3.2 สร้าง SSH Keys:

```
# สร้าง SSH key pair สำหรับ chonl
```

```
sudo -u chonl ssh-keygen -t rsa -b 4096 -C "chonl@company.com"
```

```
dev@LinuxServer:~$ sudo -u chonl ssh-keygen -t rsa -b 4096 -C "chonl@company.com"
[sudo] password for dev:
Generating public/private rsa key pair.
Enter file in which to save the key (/home/chonl/.ssh/id_rsa):
Created directory '/home/chonl/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/chonl/.ssh/id_rsa
Your public key has been saved in /home/chonl/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:BGFsflUU19mP8GUGFa7/epw79qGtnCQ2J4LefybUKWA chonl@company.com
The key's randomart image is:
+---[RSA 4096]-----+
|      .+.      o==B|
|      .o.      o .o*|
|      . ..    . o *.|
|      ..E.    + .|
|      S.. ... |
|      . o o. |
|      . ..=.o +.|
|      . . o.Bo++=|
|      . ...+=+*=|
+-----[SHA256]-----+
```

```
# Copy public key (สำหรับทดสอบ)
```

```
sudo -u chonl cp /home/chonl/.ssh/id_rsa.pub /home/chonl/.ssh/authorized_keys
```

```
sudo -u chonl chmod 600 /home/chonl/.ssh/authorized_keys
```

```
dev@LinuxServer:~$ sudo -u chonl cp /home/chonl/.ssh/id_rsa.pub /home/chonl/.ssh/authorized_keys
dev@LinuxServer:~$ sudo -u chonl chmod 600 /home/chonl/.ssh/authorized_keys
dev@LinuxServer:~$ █
```

3.3 Configure SSH Banner:

```
# สร้าง warning banner
```

```
sudo nano /etc/ssh/ssh_banner.txt
```

```
# เนื้อหา banner:
```

WARNING: Authorized access only!
All connections are monitored and recorded.
Disconnect immediately if you are not an
authorized user.

```
GNU nano 7.2
*****
WARNING: Authorized access only!
All connections are monitored and recorded.
Disconnect immediately if you are not an
authorized user.
*****
```

เพิ่มใน sshd_config

Banner /etc/ssh/ssh_banner.txt

```
PermitRootLogin no

# Authentication settings
PasswordAuthentication yes
PubkeyAuthentication yes
MaxAuthTries 3

# Idle session control
ClientAliveInterval 300
ClientAliveCountMax 2

# Allow specific users only
AllowUsers chonl nunt tuser dbuser

# Force SSH Protocol 2
Protocol 2

Banner /etc/ssh/ssh_banner.txt
```

3.4 Restart SSH และทดสอบ:

ทดสอบ config ก่อน restart

sudo sshd -t

Restart SSH service

sudo systemctl restart sshd

```
dev@LinuxServer:~$ sudo sshd -t
dev@LinuxServer:~$ sudo systemctl restart sshd
dev@LinuxServer:~$ ssh -p 2222 chonl@localhost
ssh: connect to host localhost port 2222: Connection refused
dev@LinuxServer:~$
```

ทดสอบการเชื่อมต่อ

ssh -p 2222 alice@localhost

```
dev@LinuxServer:~$ ssh -p 2222 chonl@127.0.0.1
The authenticity of host '[127.0.0.1]:2222 ([127.0.0.1]:2222)' can't be established.
ED25519 key fingerprint is SHA256:MNd8cMO/QSVNx7M/2mUsTgyYXfcmxgwKonQfzpRRL7c.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[127.0.0.1]:2222' (ED25519) to the list of known hosts.
*****
WARNING: Authorized access only!
All connections are monitored and recorded.
Disconnect immediately if you are not an
authorized user.
*****
chonl@127.0.0.1's password:
```

Task 4: Set up Firewall Rules (นายชลศักดิ์ กันทา)

4.1 Configure UFW:

Reset UFW to default

sudo ufw --force reset

```
dev@LinuxServer:~$ sudo ufw --force reset
Backing up 'user.rules' to '/etc/ufw/user.rules.20250827_110358'
Backing up 'before.rules' to '/etc/ufw/before.rules.20250827_110358'
Backing up 'after.rules' to '/etc/ufw/after.rules.20250827_110358'
Backing up 'user6.rules' to '/etc/ufw/user6.rules.20250827_110358'
Backing up 'before6.rules' to '/etc/ufw/before6.rules.20250827_110358'
Backing up 'after6.rules' to '/etc/ufw/after6.rules.20250827_110358'
```

Set default policies

sudo ufw default deny incoming

sudo ufw default allow outgoing

```
dev@LinuxServer:~$ sudo ufw default deny incoming
sudo ufw default allow outgoing
Default incoming policy changed to 'deny'
(be sure to update your rules accordingly)
Default outgoing policy changed to 'allow'
(be sure to update your rules accordingly)
dev@LinuxServer:~$
```

Allow SSH (new port)

sudo ufw allow 2222/tcp

```
dev@LinuxServer:~$ sudo ufw allow 2222/tcp
Rules updated
Rules updated (v6)
```

Allow web services

sudo ufw allow 80/tcp

sudo ufw allow 443/tcp

```
dev@LinuxServer:~$ sudo ufw allow 80/tcp
sudo ufw allow 443/tcp
Rules updated
Rules updated (v6)
Rules updated
Rules updated (v6)
```

Allow specific IPs only for SSH (optional)

sudo ufw allow from 192.168.1.0/24 to any port 2222

Enable UFW

sudo ufw enable

Show status

sudo ufw status verbose

```
dev@LinuxServer:~$ sudo ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
dev@LinuxServer:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip

To Action From
--
2222/tcp ALLOW IN Anywhere
80/tcp ALLOW IN Anywhere
443/tcp ALLOW IN Anywhere
2222/tcp (v6) ALLOW IN Anywhere (v6)
80/tcp (v6) ALLOW IN Anywhere (v6)
443/tcp (v6) ALLOW IN Anywhere (v6)
```

4.2 Advanced UFW Rules:

Rate limiting for SSH

sudo ufw limit 2222/tcp

```
dev@LinuxServer:~$ sudo ufw limit 2222/tcp
Rule updated
Rule updated (v6)
```

Allow MySQL only from specific network

sudo ufw allow from 192.168.1.0/24 to any port 3306

```
dev@LinuxServer:~$ sudo ufw allow from 192.168.1.0/24 to any port 3306
Rule added
```

Log all denied connections

sudo ufw logging on

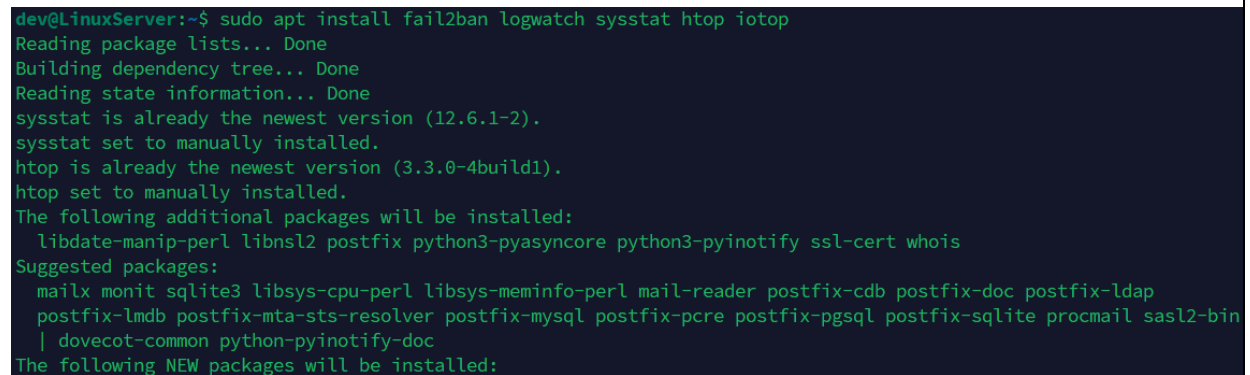
```
dev@LinuxServer:~$ sudo ufw logging on
Logging enabled
```

```
# Show numbered rules
sudo ufw status numbered
```

Task 5: Enable System Monitoring (นายชลศักดิ์ กันทา)

5.1 Install Monitoring Tools:

```
# Install required packages
sudo apt update
sudo apt install fail2ban logwatch sysstat htop iotop
```



```
# Install ELK stack components (optional)
sudo apt install elasticsearch logstash kibana
```

5.2 Configure Fail2Ban:

```
# Backup original config
sudo cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.conf.backup
```



```
# สร้าง local config
sudo nano /etc/fail2ban/jail.local
```

เนื้อหาไฟล์:

```
[DEFAULT]
```

```
bantime = 3600
findtime = 600
maxretry = 3
backend = systemd

[sshd]
enabled = true
port = 2222
logpath = /var/log/auth.log
maxretry = 3
bantime = 3600

[apache-auth]
enabled = true
port = http,https
logpath = /var/log/apache2/error.log

[apache-badbots]
enabled = true
port = http,https
logpath = /var/log/apache2/access.log
bantime = 86400
maxretry = 1
```

5.3 Configure System Monitoring:

```
# Enable sysstat
sudo systemctl enable sysstat
sudo systemctl start sysstat
```

```
dev@LinuxServer:~$ sudo systemctl enable sysstat
sudo systemctl start sysstat
Synchronizing state of sysstat.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable sysstat
dev@LinuxServer:~$
```

Create monitoring script

sudo nano /usr/local/bin/system_monitor.sh

#!/bin/bash

System monitoring script

DATE=\$(date)

echo "=== System Monitor Report - \$DATE ===" >> /var/log/system_monitor.log

CPU Usage

echo "CPU Usage:" >> /var/log/system_monitor.log

top -bn1 | grep "Cpu(s)" >> /var/log/system_monitor.log

Memory Usage

echo "Memory Usage:" >> /var/log/system_monitor.log

free -h >> /var/log/system_monitor.log

Disk Usage

echo "Disk Usage:" >> /var/log/system_monitor.log

df -h >> /var/log/system_monitor.log

Active Users

echo "Active Users:" >> /var/log/system_monitor.log

who >> /var/log/system_monitor.log

Failed Login Attempts

echo "Recent Failed Logins:" >> /var/log/system_monitor.log

tail -10 /var/log/auth.log | grep "Failed password" >> /var/log/system_monitor.log

```
echo "======" >> /var/log/system_monitor.log
```

```
GNU nano 7.2 /usr/local/bin/system_monitor.sh
#!/bin/bash
# System monitoring script
DATE=$(date)
echo "=== System Monitor Report - $DATE ===" >> /var/log/system_monitor.log

# CPU Usage
echo "CPU Usage:" >> /var/log/system_monitor.log
top -bn1 | grep "Cpu(s)" >> /var/log/system_monitor.log

# Memory Usage
echo "Memory Usage:" >> /var/log/system_monitor.log
free -h >> /var/log/system_monitor.log

# Disk Usage
echo "Disk Usage:" >> /var/log/system_monitor.log
df -h >> /var/log/system_monitor.log

# Active Users
echo "Active Users:" >> /var/log/system_monitor.log
who >> /var/log/system_monitor.log

# Failed Login Attempts
echo "Recent Failed Logins:" >> /var/log/system_monitor.log
tail -10 /var/log/auth.log | grep "Failed password" >> /var/log/system_monitor.log

echo "======" >> /var/log/system_monitor.log
```

```
# Make executable
```

```
sudo chmod +x /usr/local/bin/system_monitor.sh
```

```
dev@LinuxServer:~$ sudo chmod +x /usr/local/bin/system_monitor.sh
dev@LinuxServer:~$
```

```
# Add to crontab (run every hour)
```

```
sudo crontab -e
```

```
# เพิ่มบรรทัด:
```

```
0 * * * * /usr/local/bin/system_monitor.sh
```

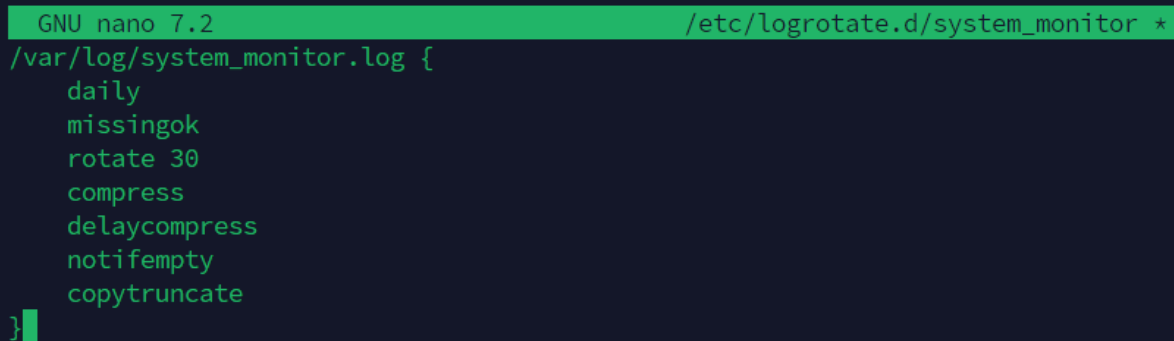
5.4 Configure Log Rotation:

```
# Create logrotate config
```

```
sudo nano /etc/logrotate.d/system_monitor
```



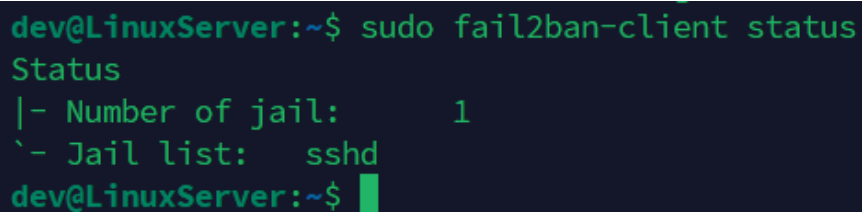
```
/var/log/system_monitor.log {  
    daily  
    missingok  
    rotate 30  
    compress  
    delaycompress  
    notifempty  
    copytruncate  
}
```



```
GNU nano 7.2 /etc/logrotate.d/system_monitor *  
/var/log/system_monitor.log {  
    daily  
    missingok  
    rotate 30  
    compress  
    delaycompress  
    notifempty  
    copytruncate  
}
```

ที่ต้องจับภาพ:

```
sudo fail2ban-client status
```



```
dev@LinuxServer:~$ sudo fail2ban-client status  
Status  
|- Number of jail:      1  
`- Jail list:  sshd  
dev@LinuxServer:~$
```

```
sudo fail2ban-client status sshd
```

```
dev@LinuxServer:~$ sudo fail2ban-client status sshd
Status for the jail: sshd
|- Filter
| |- Currently failed: 0
| |- Total failed: 0
| `-- Journal matches: _SYSTEMD_UNIT=sshd.service + _COMM=sshd
`-- Actions
    |- Currently banned: 0
    |- Total banned: 0
    `-- Banned IP list:
dev@LinuxServer:~$ █
```

ไฟล์ /var/log/system_monitor.log

sudo systemctl status fail2ban

```
dev@LinuxServer:~$ sudo systemctl status fail2ban
• fail2ban.service - Fail2Ban Service
   Loaded: loaded (/usr/lib/systemd/system/fail2ban.service; enabled; preset: enabled)
   Active: active (running) since Wed 2025-08-27 11:22:16 UTC; 11min ago
     Docs: man:fail2ban(1)
   Main PID: 5074 (fail2ban-server)
      Tasks: 5 (limit: 4605)
     Memory: 24.6M (peak: 25.1M)
        CPU: 2.507s
    CGroup: /system.slice/fail2ban.service
            └─5074 /usr/bin/python3 /usr/bin/fail2ban-server -xf start

Aug 27 11:22:16 LinuxServer systemd[1]: Started fail2ban.service - Fail2Ban Service.
Aug 27 11:22:16 LinuxServer fail2ban-server[5074]: 2025-08-27 11:22:16,816 fail2ban.configreader [5074]: WARNING 'a
Aug 27 11:22:17 LinuxServer fail2ban-server[5074]: Server ready
lines 1-14/14 (END)
```