Title: Ubuntu Lab Auto-Restore & Google Form IP-Restricted Access

Table of Contents

1. Deep Freeze Alternatives on Ubuntu

fsprotect

```
sudo apt update
sudo apt install fsprotect
sudo fsprotect /home
```

overlayroot

```
sudo apt install overlayroot
sudo nano /etc/overlayroot.conf
# Add the line: overlayroot=tmpfs
```

Timeshift

```
sudo apt install timeshift
sudo timeshift --create --comments "Clean Snapshot"
sudo timeshift --restore
```

2. Auto-Restore Snapshot on Reboot and Scheduling

Auto-Reboot via Cron

```
sudo crontab -e
# Add line to reboot daily at 2 AM\ n0 2 * * * /sbin/reboot
```

Timeshift Scheduled Restore

```
sudo timeshift --list
sudo crontab -e
# Add line to restore specific snapshot at 2 AM
0 2 * * * timeshift --restore --snapshot "2025-11-10_09-00-00" --yes
```

3. Google Form IP-Restricted Access Using NGINX

```
server {
    listen 80;
    server_name localhost;
    set $allowed_ip "192.168.1.25";

    location /secureform {
        if ($remote_addr != $allowed_ip) {
            return 403;
        }
        return 302 https://docs.google.com/forms/d/e/1FAIpQLSfEXAMPLEFORM/
viewform;
    }

    error_page 403 /403.html;
    location = /403.html {
        internal;
        add_header Content-Type text/html;
        return 200 "<h1>Access Denied</h1><p>This form is not available on your
IP.</p>";
    }
}
```

Reload NGINX:

```
sudo nginx -t
sudo systemctl reload nginx
```

4. Multi-PC Google Form Access

```
server {
    listen 80;
    server_name localhost;

    allow 192.168.1.10;
    allow 192.168.1.15;
    allow 192.168.1.20;
    deny all;

    location /secureform {
        return 302 https://docs.google.com/forms/d/e/1FAIpQLSfEXAMPLEFORM/
viewform;
    }

    error_page 403 /403.html;
    location = /403.html {
        internal;
        add_header Content-Type text/html;
        return 200 "<h1>Access Denied</h1><p>You are not authorized to open
this form.</p>";
    }
}
```

5. Centralized Server + 120 Clients Auto-Sync & Overlayroot Setup

Server Setup

```
sudo apt update
sudo apt install rsync nginx -y
sudo mkdir -p /srv/labimage
sudo chmod -R 755 /srv/labimage
hostname -I
```

Client Auto-Sync Script

```
sudo nano /usr/local/bin/lab-sync.sh
```

```bash
#!/bin/bash
SERVER_IP="192.168.1.10"
REMOTE_DIR="/srv/labimage/"
LOCAL_DIR="/opt/labimage/"

mkdir -p "$LOCAL_DIR"

SERVER_HASH=$(rsync --dry-run --checksum -avz "$SERVER_IP:$REMOTE_DIR"
"$LOCAL_DIR" | md5sum | cut -d" " -f1)
LOCAL_HASH_FILE="/opt/labimage/.lastsync"

if [ ! -f "$LOCAL_HASH_FILE" ] || [ "$SERVER_HASH" != "$(cat
$LOCAL_HASH_FILE)" ]; then
    echo "Updating from server..."
    rsync -avz --delete "$SERVER_IP:$REMOTE_DIR" "$LOCAL_DIR"
    echo "$SERVER_HASH" > "$LOCAL_HASH_FILE"
    echo "Update complete at $(date)"
fi
```

```
sudo chmod +x /usr/local/bin/lab-sync.sh
sudo crontab -e
# */15 * * * * /usr/local/bin/lab-sync.sh >> /var/log/lab-sync.log 2>&1
```

Overlayroot Installation on Clients

```
sudo apt install overlayroot -y
echo "overlayroot=tmpfs" | sudo tee /etc/overlayroot.conf
```

---

6. Sample Scripts for Server and Clients

- **server-setup.sh** → configure `/srv/labimage` and NGINX for updates
- **client-setup.sh** → configure auto-sync every 15 min, overlayroot, and cron

Scripts contain variables for server IP, client IPs, and directories.

---

7. Notes & Best Practices

- Ensure lab PCs have **static IPs** for reliable IP restrictions.
- Use **Gigabit LAN** for 120 clients to avoid congestion.
- Enable logging to monitor sync status (`/var/log/lab-sync.log`).
- Overlayroot ensures all clients auto-reset on reboot (Deep Freeze style).
- For large image deployment, consider **Clonezilla SE** for PXE boot mass restore.

End of Document