Day 1: User Creation

Problem Statement:
Create a user named ravi on App Server 2 in Stratos Datacenter. Set the expiry date to 2024-03-28.

Solution:
1. SSH into App Server 2: ssh steve@stapp02
2. Switch to root: sudo -i
3. Create user with expiry: useradd -e 2024-03-28 ravi
4. Verify user created: getent passwd ravi
5. Set password for login: passwd ravi

Command Explanations:
useradd -e 2024-03-28 ravi → Create user with expiry date 2024-03-28
getent passwd ravi → Check if user ravi exists in system user database
passwd ravi → Set a password for the user ravi

Day 2: Disable Root SSH Login

Problem Statement:
Disable direct SSH root login on all app servers.

Solution:
1. SSH into the server: ssh tony@stapp01
2. Edit SSH config: sudo vi /etc/ssh/sshd_config
3. Find PermitRootLogin yes and change to PermitRootLogin no
4. Save file and restart SSH: sudo systemctl restart sshd
5. Verify by trying: ssh root@stapp01

Command Explanations:
vi /etc/ssh/sshd_config → Open SSH configuration file for editing
PermitRootLogin no → Disable SSH login directly as root
systemctl restart sshd → Restart SSH service to apply changes

Day 3: File Permission

Problem Statement:
Grant executable permissions to the /tmp/xfusioncorp.sh script on App Server 3 for all users.

Solution:
1. SSH into App Server 3: ssh banner@stapp03
2. Switch to root: sudo -i
3. Change permission: chmod a+x /tmp/xfusioncorp.sh
4. Verify: ls -l /tmp/xfusioncorp.sh

Command Explanations:
chmod a+x file → Give execute permission to all users
ls -l file → List file details including permissions

Day 4: Git Repository

Problem Statement:
Install git and create a bare repository at /opt/blog.git on the Storage server.

Solution:
1. SSH into storage server: ssh natasha@ststor01
2. Switch to root: sudo -i

3. Install git: yum install -y git
4. Create bare repo: git init --bare /opt/blog.git
5. Verify: ls -ld /opt/blog.git

Command Explanations:
yum install -y git → Install git package
git init --bare /opt/blog.git → Initialize an empty bare repository
ls -ld /opt/blog.git → Verify repository directory exists

Day 5: SELinux

Problem Statement:
Install SELinux packages and permanently disable SELinux until maintenance reboot.

Solution:
1. Install SELinux tools: yum install -y policycoreutils selinux-policy
2. Edit config: sudo vi /etc/selinux/config
3. Set SELINUX=disabled
4. Verify file saved. Reboot is planned so no need now.

Command Explanations:
yum install -y policycoreutils selinux-policy → Install SELinux tools
/etc/selinux/config → Main SELinux config file
SELINUX=disabled → Disable SELinux permanently

Day 6: Cron Jobs

Problem Statement:
Install cronie and configure cron to echo hello every 5 minutes into /tmp/cron_text.

Solution:
1. SSH into app servers: ssh tony@stapp01 (repeat for others)
2. Switch to root: sudo -i
3. Install cronie: yum install -y cronie
4. Start service: systemctl enable --now crond
5. Add cron job: echo '*/5 * * * * echo hello > /tmp/cron_text' >> /var/spool/cron/root
6. Verify after 5 minutes: cat /tmp/cron_text

Command Explanations:
yum install -y cronie → Install cron job scheduler package
systemctl enable --now crond → Start and enable cron service
cat /tmp/cron_text → Check if cron wrote hello into file

Day 7: Passwordless SSH

Problem Statement:
Set up passwordless SSH from thor@jump_host to all app servers via sudo users.

Solution:
1. On jump host, generate keys: ssh-keygen -t rsa -b 4096
2. Copy key to server: ssh-copy-id tony@stapp01
3. Repeat for steve@stapp02 and banner@stapp03
4. Verify: ssh tony@stapp01 (should not ask password)

Command Explanations:
ssh-keygen -t rsa -b 4096 → Generate SSH key pair
ssh-copy-id user@host → Copy public key to remote server

ssh user@host → Login without password using key authentication