

Backups - exercises

We are going to do something similar to what was done in the slides, i.e.: take a backup of /etc and /usr/local/etc to ANOTHER machine in the class, effectively implementing a "remote backup".

Required knowledge - you need to understand:

- manipulating ssh
- also, understanding ssh keys (not mandatory)

Prerequisites:

- you need to have an account on the machine you will backup TO

1. **Find a partner in the class, with whom you will be doing backups**
2. **Agree with this person which account you will use -- ideally, create a "backup" user, which you can create with the "pw" user:**

```
# pw user add backup -d /home/backup -m -s /bin/sh
# passwd backup
```

3. **If you understand SSH keys sufficiently, follow these instructions, otherwise skip to step 4.**

You will copy your .ssh/id_dsa.pub key file to the ".ssh/authorized_keys" of the account on the REMOTE machine, i.e.: if the machine you will be working with is "pc30", and YOUR machine is "pc40" then you will need:

- an account ("backup") on pc30
- a generated, passphrase-less key on YOUR machine (pc40) in .ssh/id_dsa.pub (remember, that file is in your HOME directory -- under .ssh/)
- a copy of this key on the account ("backup") on the REMOTE ("pc30") machine

4. **Using what you have learned during the presentation, we will attempt to backup the /etc and /usr/local/etc.**

We do this as root.

```
# rsync -avzR /etc /usr/local/etc backup@<remotemachine>:
```

... where <remotemachine> is the name or IP address of the REMOTE machine you have created the account on.

5. **Observe what happens, and log on to the remote machine to see that the backup has worked:**

```
# ssh backup@<remotemachine>
...
% ls -l
```

6. **Did it work ? :)**

7. **You could then use a script (program) to move the backup out of the way, and repeat this backup process automatically once every day, or twice a day, for example.**