# TEST PROJECT
# IT NETWORK SYSTEM ADMINISTRATION

## WSG23_TP39_Module-A: Open Source Environment

Developed by:

Kai Schell
Roman Steinhart

Date: 13.06.2023

Version: 1.0

# Übersicht

# 1 INTRODUCTION

The competition has a fixed start and finish time. You must decide how to best divide your time.

**Please carefully read the following instructions!**

When the competition time ends, please leave your station in a running state. The assessment will be done in the state as it is. No reboot will be initiated as well as powered off machines will not be powered on!

Please use the information in the appendix to access your virtual machines as well as the virtual machines themselves.

## 1.1 System configuration

All virtual machines are configured as follows:

Region/timezone: Europe/Warsaw

Locale: English US (UTF-8)

Key Map: English US

IP address: dhcp

Operating System: Servers: Debian 12, Clients: Linux Mint

Every VM you see in the topology chart at the end of this document is preinstalled on the physical host, named accordingly. The hosts use Proxmox VE 7.4 as a virtualisation platform and you can connect to the resources using your browser.

The Proxmox webinterface is reachable on https://module-a.skill39.de:8006 . Your credentials are listed in the appendix.

The preinstalled VMs contain only the base system and few additional packages, you can install additional packages as you want.

All virtual machines have internet access by default and the public debian apt repositories configured. The router vm acts as gateway to the internet and DHCP server. It is preconfigured, you are not supposed to make any configuration changes to it. In case you need to troubleshoot something you can find the credentials in the appendix

The test project does not always give you an exact specification. In these situations you have the chance to choose which software to use, which path to follow - you will find information at the tasks about the paths you can choose from. The more sophisticated a solution is the better mark you are going to get for it.

## 1.2 Instructions to the competitor

You have three hours to complete the task.

All work steps / your entire procedure are to be roughly documented by you in a file (~/docs/skills-documentation.txt) on the client computer. The documentation can be made in bullet points and should contain the relevant steps, reasons for certain decisions, but also all possibly occurring problems and other conspicuities.

If you get stuck at a certain point in your work, please also record this in your documentation. In this case you can get help from a colleague (i.e. the jury). The use of help will be documented by both you and the jury and a certain number of points will be deducted for this, depending on the help used.

Read all tasks thoroughly before starting your work.

## 1.3 Further notes

- If you get stuck at certain points, remember that Linux provides extensive documentation: In the form of help options, man pages, the info system, package-specific documentation, etc.
- (or just use Google)
- We recommend that you do not manually modify or delete the log & history files. They serve the jury, in addition to your documentation, as an aid in the evaluation.

# 2 DESCRIPTION OF PROJECT AND TASKS

## 2.1 Situation

You work at the company Floss-Skills. When your supervisor asks you to upload a certain file to DropBox, you suggest installing the free DropBox and Google Services alternative Nextcloud on a company server instead.

You should also perform a backup of the most important data on the server, according to your company's policies. The backup should be automated, so that the backups are transferred to a backup server.

Since you are also familiar with other file systems apart from the standard Linux ext4 file system and use your knowledge to move the Nextcloud data directory to a partition with xfs so that you can comfortably create snapshots with the corresponding XFS tools.

# 3 TASKS

## 3.1 Preparing mint-client

1. Configure IP and hostname according to the tables in the appendix.

### 3.1.1 SSH key

1. To prepare the passwordless login on the server, create an SSH key pair on your client machine as user `aflosser` with the command `ssh-keygen` for login on the web server. The name of the key should be `web.floss-skills.xyz`, be of type `ecdsa`, have a length of `384` bits and be secured with the password `webserverssh`.

2. Then connect to the root account on the web server via SSH and authenticate with the password of root user.

### 3.1.2 Easy SSH access

1. To be able to log on to the web-server more comfortably in the future, create a user-specific SSH client configuration in the file `~/.ssh/config`:

2. The login via SSH as user `awebber` on the web server `web.floss-skills.xyz` should be possible by entering `ssh nc-server` as `aflosser` with the created SSH key. Don't use shell aliases to accomplish this task.

## 3.2 Preparing web-server

### 3.2.1 User, sudo and SSH

1. Create a regular user on the webserver with the username `awebber` and the full name Alex Webber. The user should use BASH as the login shell and be given its own home directory under `/home` with the name `awebber`. The home directory should also contain the "default files" from the skeleton directory. The password should be `awebberpw`.

2. Now install `sudo` and add your newly created user to the appropriate group so that he can gain root privileges by issuing the `sudo` command and entering his password.

3. **Important**: All further tasks that require root privileges should be performed by prefixing `sudo` without starting a permanent root shell!

4. Now transfer the public part of the previously created SSH key to the `awebber` user on the server so that you can log in without specifying a password.

5. Übertragen Sie nun den öffentlichen Teil des zuvor erstellten SSH Schlüssels auf den Server, so dass Sie sich ohne Angabe eines Passworts anmelden können.

6. Then perform the following configuration of the SSH server:

    1. The `root` user is no longer allowed to log in to the system via SSH.

    2. Login via password is to be prohibited in general.

7. Test if everything works as desired, not that you lock yourself out by mistake! Make a note about it in your documentation, how you made sure that this does not happen. Then apply these changes.

### 3.2.2 IP configuration and FQDN

1. Configure IP and hostname according to the tables in the appendix.

2. Also do any further configuration steps so that the server is connected to the network afterwards.

3. An additional IP address (10.0.99.90) is to be added to the webserver, under which the Nextcloud instance will be accessible. Configure the additional IP address specified above for the Nextcloud on the same interface.

4. Change the hostname of the web server to `web`.

5. Note: After the IP configuration, do a reboot and no ifup / ifdown or `sytemctl restart networking.service`. Otherwise, you may not be able to connect to the server. If this does happen, you will probably have to contact your colleagues at the data center (jury).

## 3.3 Installing Nextcloud

**Note**: The installation of Nextcloud should be done manually, i.e. not via e.g. a package from the repos, an installation script or a container solution like docker!

1. Transfer the Nextcloud tarball from your clients `Downloads` directory to the web server and then unpack it into the directory `/var/www/nextcloud`.

2. Now install the Apache2 web server, PHP version 8.2, the appropriate library so that the web server can also serve PHP files (`libapache2-mod-php`), the MariaDB database management system and the other PHP modules required for Nextcloud. Your colleague has left you the following list for this purpose: `curl`, `mbstring`, `intl`, `gmp`, `bcmath`, `xml`, `imagick`, `gd`, `zip`, `mysql`

3. Now create a MariaDB database for Nextcloud with the name `nextcloud_db`.

4. Create the MariaDB user cloudius with the password `nextcloud_db_pw`.

5. The user should have full access to the complete database `nextcloud_db`.

6. Now you need to create a VirtualHost configuration for the web server named `cloud.floss-skills.xyz.conf` with the following content:

1. Nextcloud should only be accessible via SSL, so requests to port 80 should be redirected to port 443.

2. Nextcloud should be reachable via the IP `10.100.99.90` (additionally configured on the interface)under the hostname `cloud.floss-skills.xyz`.

3. Further, ensure that all regular requests are logged to the `/var/log/apache2/cloud.floss-skills.xyz_access.log` file, and all errors are logged to the `/var/log/apache2/cloud.floss-skills.xyz_error.log` file.

4. For the SSL connection, temporarily create a self-signed Snakeoil certificate for testing. Use the default template given in the manpage/documentation. The certificate should be named `ssl-cloud.floss-skills.xyz.crt` and be valid for the URL `cloud.floss-skills.xyz`.

7. Nextcloud's data partition should be on a separate XFS partition for convenient dumping of snaphots. Fortunately, the basic partitioning was done with LVM, so you can create another 10G logical volume with the name ncdata and the XFS file system without any problems. Mount the new partition statically in the `/etc/fstab` file so that it is automatically mounted at boot time. Continue to use the default mount options. The partition should be mounted by its UUID and not by the device file.

## 3.4 SSH and local DNS on the client

1. On the client, adjust the local DNS so that Nextcloud is accessible under the Nextcloud IP at the URL `cloud.floss-skills.xyz` and the web server is accessible under its own IP and the name `web.floss-skills.xyz`. Ensure that calling `https://cloud.floss-skills.xyz` in the browser of mint-client opens nextcloud.

2. Carry out any steps that may still be necessary so that Nextcloud is successfully installed and the Nextcloud user `nexti`, which is to be created during the installation, can log in there with the password `nextipw`. No further configuration of Nextcloud is necessary.

3. Also, make sure that the permissions on Nextcloud files are set as securely as possible, i.e. that the web server, for example, is only allowed to write to the files where this is really necessary.

## 3.5 Backup script on the web-server

1. Create a backup script named `back-it-up.sh` in `/root/scripts`.

2. The directories `/home`, `/etc`, `/root` and `/var` shall be backed up, all file attributes like ownership, permissions and timestamps shall be kept.

3. The subdirectory `/var/log` should be excluded from the backup, since the logs are also stored on a central logging server. Note: Make sure that there are no other unnecessary files in the backups later.

4. The backup should be done using `tar` and the archive should be compressed using `bzip2`.

5. The backups should be named according to a certain naming scheme, so that the hostname and a current timestamp are included: `backup_<hostname>_<timestamp>.tar.bz2` Since you may want to use this script in future on other servers too properties like hostname and timestamp should not be hardcoded.

6. The timestamp should have the format `YYYYMMDDhhmm` (date in american format, 24 hour time).

7. The backups should be encrypted with GPG and signed.

    1. For this you need to create a GPG key pair as `awebber` on the server with the following specifications:

        1. Full name: Alex Webber

        2. Email: `awebber@floss-skills.xyz`

        3. PW: `awebber_gpg`

        4. For all other properties take the default values offered by the OS.

8. The backup files should be transferred to `/backups` on `backup-server.` It's up to you on how the backups are transferred to the backup-server.

9. The script should be executed every Saturday morning at 2:32.

10. Make sure that the `root` user and the `awebber` user can call the script without specifying a path.

## 3.6 backup-server and NFS

1. Configure IP and hostname according to the tables in the appendix.

2. Expose the directory `/nfsdata` as NFS share. Only `mint-client` should be allowed to access the NFS share.

3. Secure the NFS share with industry standard measures.

4. Mount the NFS share on the mint-client.

# 4 APPENDIX

## 4.1 Users, IP addresses, Hostnames

### 4.1.1 mint-client

| username | password |
|---|---|
| root | rootpw |
| aflosser | aflosserpw |

| current IP | designated IP |
|---|---|
| dhcp | dhcp |

| current hostname | designated hostname |
|---|---|
| mint-client | mint |

### 4.1.2 web-server

| username | password |
|---|---|
| root | rootpw |
| awebber | awebber |

| current IP | designated IPs |
|---|---|
| dhcp | 10.0.99.10 |
| | 10.0.99.90 (Nextcloud) |

| current hostname | designated hostname |
|---|---|
| web-server | web |

### 4.1.3 backup-server

| username | password |
|---|---|
| root | rootpw |
| abackupper | abackupperpw |

| current IP | designated IP |
|---|---|
| dhcp | 10.0.99.20 |

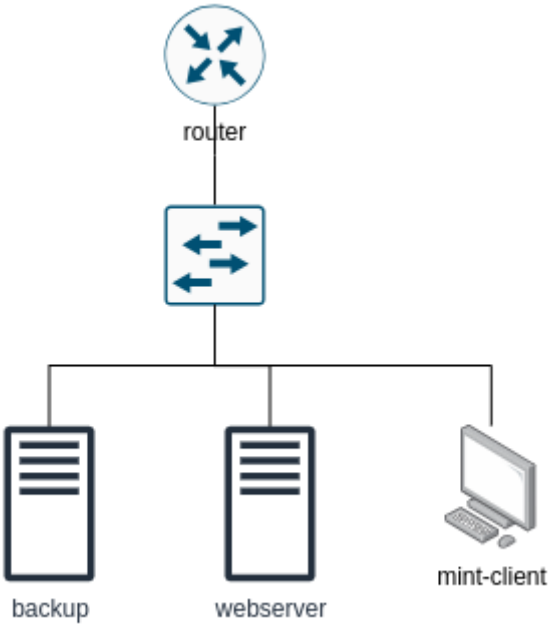
| current hostname | designated hostname |
|---|---|
| backup-server | backup |

### 4.1.4 router

| username | password |
|---|---|
| root | rootpw |

| current IP | designated IP |
|---|---|
| 10.0.99.1 | 10.0.99.1 |

## 4.2 Topology




## 4.3 Proxmox access

| | |
|---|---|
| Firstname | Max |
| Lastname | Guhlke |
| Proxmox URL | https://module-a.skill39.de:8006 |
| Username | `competitor-10` |
| Password | `7MEeDzua` |
| Realm | Proxmox VE authentication server |