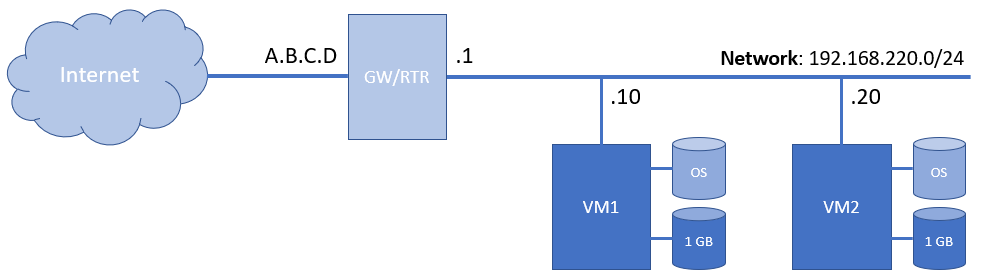
# Exam: LSA (2021.04) – 2021.06.20

### Infrastructure & rules

You will have to accomplish a set of tasks in the following infrastructure:



All exam machines (VM1 and VM2) are of the same type (in terms of OS and hardware parameters)

Pease note that if you are given IP address of **A.B.C.D** then the following port-forwarding rules are in place:

|  |  |  |  |
| --- | --- | --- | --- |
| VM1 | A.B.C.D:10022 -> 192.168.220.10:22 | VM2 | A.B.C.D:20022 -> 192.168.220.20:22 |
| A.B.C.D:10080 -> 192.168.220.10:80 | A.B.C.D:20080 -> 192.168.220.20:80 |

Be sure to **follow strictly** the **naming** **conventions** (users, groups, folders, files, etc.) specified in the tasks checklist

Tasks execution order **should not be derived** from the order in which they are listed below. However, please note that there are tasks that depend on the successful completion of one or more other tasks

Usually, all steps could be achieved following different paths and using different tools. In the end, not the means, but **the** **end results are being measured**

All changes should remain there after system reboot. For example, new file systems should be mounted also in **/etc/fstab**. The same applies for all services - they should be **operational after system reboot**

Please note that the **/data** folder and all its sub-folders do not exist (on both VMs), thus you must create them 😉

Note also, that if a utility like **tar**, **xz**, or any other, needed for the successful completion of the tasks is missing, then you are expected to install it 😊

### Tasks

#### Disks and File Systems - 9 tasks, 13 pts

* (T101, 2 pts) **VM1:** Use the appropriate tool **to create a new** primary partition using the **MBR** partitioning scheme on the **smaller (1 GB) and empty hard disk** drive with size of **500 MB** and type set to **Linux LVM**
* (T102, 1 pts) **VM1:** Create a physical volume on the new partition, created earlier
* (T103, 1 pts) **VM1:** Create a volume group named **vg\_exam** on the new physical volume
* (T104, 1 pts) **VM1:** Create a logical volume named **lv\_exam** on the new volume group (use 100% of the available space in the volume group)
* (T105, 1 pts) **VM1:** Create an **ext4** file system on the **lv\_exam** logical volume
* (T106, 2 pts) **VM1:** Mount the new file system on the **/data/ext4** folder and add a record in the **/etc/fstab** file
* (T107, 2 pts) **VM2:** Use the appropriate tool **to create a new** partition using the **GPT** partitioning scheme on the  **smaller (1 GB) and empty hard drive** with size of **500 MB** and type set to **Linux Filesystem**
* (T108, 1 pts) **VM2:** Create **xfs** file system on the new partition
* (T109, 2 pts) **VM2:** Mount the new file system on the **/data/xfs** folder and add a record in the **/etc/fstab** file

#### Directories and Files - 8 tasks, 13 pts

|  |  |
| --- | --- |
| * (T201, 2 pts) **VM1:** Create series of directories under the path **/data/projects** with the following structure (refer to the image) * (T202, 1 pts) **VM1:** In each folder **documents** from the previous step create an **empty** file named **readme.txt** * (T203, 1 pts) **VM1:** In each folder **source** create file named **code.sh** that contains just the command (as text) **whoami** | /data  └── projects  ├── project1  │ ├── documents  │ └── source  └── project2  ├── documents  └── source |

* (T204, 2 pts) **VM1:** Create a file named **unique-animals.txt** in the folder **/data/animals** that contains the **sorted list in reverse order** of the **unique animals** (just their names) found in the **/important/animal-stories.txt** file
* (T205, 2 pts) **VM1:** Create a **xz** compressed archive named **important-bak.tar.xz** of the **/important** folder and its content and store it in the **/data/archive** folder
* (T206, 2 pts) **VM2:** Create a text file **exam\_files.txt** (and store it under **/data/find** folder) that contains the sorted (in ascending order) list of all the places (full path, including the name) where files with **exam.lsa** name are found
* (T207, 1 pts) **VM2:** Create a new file based on the **exam\_files.txt** with all words (in the file) **exam** changed to **EXAM** and store it as **exam\_files\_upper.txt** in the same folder (**/data/find**)
* (T208, 2 pts) **VM2:** Create a copy of the **/important/animal-stories.txt** file as **/data/animals/tigers.txt** file which contains only lines that contain the **tiger** text no matter the register (size of the letters) or position in a sentence

#### Users and Permissions - 5 tasks, 7 pts

* (T301, 2 pts) **VM1:** Create a user **ivan** with full name **Ivan Petkov**, with **some password** and auto-created **home folder**
* (T302, 2 pts) **VM1:** Create a user **mariana** with full name **Mariana Parusheva**, with **some password** and auto-created **home folder**
* (T303, 1 pts) **VM1:** Create a group named **team**
* (T304, 1 pts) **VM1:** Make both users, **ivan** and **mariana**, part of the **team** group
* (T305, 1 pts) **VM1:** Make user **ivan** and group **team** owners of the **/data/projects** folder and all its **sub-folders** and **files**

#### Software and Services - 6 tasks, 8 pts

* (T401, 1 pts) **VM1:** Install **Apache2**, start it, and enable it to run on boot
* (T402, 2 pts) **VM1:** Install **PHP** and any additional packages required to connect to ***MariaDB***
* (T403, 1 pts) **VM1:** Install **MariaDB** (client and server), start it, and enable it to run on boot
* (T404, 1 pts) **VM2:** Install **NGINX**, start it, and enable it to run on boot
* (T405, 1 pts) **VM2:** Download the appropriate package for your distribution:
  + For ***CentOS*** and ***openSUSE***, download: [**https://zahariev.pro/linux/hello-lsa/releases/hello-lsa-1.0-1.el8.x86\_64.rpm**](https://zahariev.pro/linux/hello-lsa/releases/hello-lsa-1.0-1.el8.x86_64.rpm)
  + For ***Ubuntu***, download: [**https://zahariev.pro/linux/hello-lsa/releases/hello-lsa-1.0\_amd64.deb**](https://zahariev.pro/linux/hello-lsa/releases/hello-lsa-1.0_amd64.deb)
* (T406, 2 pts) **VM2:** Install the downloaded package (do not delete the downloaded file)

#### Web and DB - 9 tasks, 11 pts

* (T501, 1 pts) **VM1:** Secure the **MariaDB** instance by setting the **root** database user password to **Password1**
* (T502, 1 pts) **VM1:** Using a tool like **wget** or **curl** download the following file [**https://zahariev.pro/exam/lsa/web.tar.gz**](https://zahariev.pro/exam/lsa/web.tar.gz)and store it in the home folder of the **exam** user
* (T503, 1 pts) **VM1:** Extract the contents of the downloaded file in the home folder of the **exam** user (a new folder **webapp** must appear). **Do not delete** the downloaded file
* (T504, 1 pts) **VM1:** Using the extracted file (**~/webapp/db.sql**) and the **mysql** tool, restore (create) the database (**performancedb**)
* (T505, 1 pts) **VM1:** Using the **mysql** tool, create a **perfdbuser** database user with password set to **Password1** who has full privileges on the **performancedb** (or at least can connect to it) database
* (T506, 2 pts) **VM1:** Insert **only one control record** in the **performancetbl** table with the following values (**proc\_num**, **-99**)
* (T507, 1 pts) **VM1:** **Copy** the web application file (**~/webapp/app.php**) to the **/var/www/html** (or the default document root folder, depending on your distribution of choice) folder and **rename** it to **performance.php**
* (T508, 2 pts) **VM1:** Edit the **performance.php** file, change the values to match yours (the database user created in ***T505*** and its password) and save it
* (T509, 1 pts) **VM2:** Create a static html page named **exam.html** in the **default** web documentfolder (depending on your distribution of choice) that contains your **username** in the **SoftUni** portal

#### Scripting and Schedules - 4 tasks, 6 pts

* (T601, 1 pts) **VM1:** Create a script file named **performance.sh** in the **/data/scripts** folder
* (T602, 1 pts) **VM1:** Change permissions of the **/data/scripts/performance.sh** file to **executable** for **everyone**
* (T603, 3 pts) **VM1:** When executed the **performance.sh** script should capture **the number of running processes** and **store the results** in the **performancetbl** table from the previous set of tasks
* (T604, 1 pts) **VM1:** **Schedule** the script for the **exam** user to execute **every two minutes**

#### Networking and Connectivity - 2 tasks, 2 pts

* (T701, 1 pts) **VM1:** Make sure that the web server is accessible from outside the virtual machine by adjusting the firewall if needed
* (T702, 1 pts) **VM2:** Make sure that the web server is accessible from outside the virtual machine by adjusting the firewall if needed