# M8: Exam Preparation (Sample Exam)

### Infrastructure & rules

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

Chart

Description automatically generated

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:10001 -> 192.168.10.10:22 | VM2 | A.B.C.D:10002 -> 192.168.10.20:22 |

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** except stated otherwise

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 😊

And one last reminder, you **MUST** use the **exam** user to execute the tasks. Should you need administrative access, use the **sudo** command *(no password would be required)*

### Tasks

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

* [(T101, 2 pts)](http://192.168.81.160/control.php#t101) **VM1**: Use the appropriate tool **to create a new** primary partition using the **MBR** partitioning scheme on the **smaller (2 GB) and empty hard disk** drive with size of **800 MB** and type set to **Linux LVM**
* [(T102, 1 pts)](http://192.168.81.160/control.php#t102) **VM1**: Create a physical volume on the new partition, created earlier
* [(T103, 1 pts)](http://192.168.81.160/control.php#t103) **VM1**: Create a volume group named **vg\_exam** on the new physical volume
* [(T104, 1 pts)](http://192.168.81.160/control.php#t104) **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)](http://192.168.81.160/control.php#t105) **VM1**: Create an **xfs** file system on the **lv\_exam** logical volume
* [(T106, 2 pts)](http://192.168.81.160/control.php#t106) **VM1**: Mount the new file system on the **/data/xfs** folder and add a record in the **/etc/fstab** file
* [(T107, 2 pts)](http://192.168.81.160/control.php#t107) **VM2**: Use the appropriate tool **to create a new** partition using the **MBR** partitioning scheme on the **smaller (2 GB) and empty hard drive** with size of **650 MB** and type set to **Linux Filesystem**
* [(T108, 1 pts)](http://192.168.81.160/control.php#t108) **VM2**: Create **ext4** file system on the new partition
* [(T109, 2 pts)](http://192.168.81.160/control.php#t109) **VM2**: Mount the new file system on the **/data/ext4** folder and add a record in the **/etc/fstab** file

#### Directories and Files - 9 tasks, 16 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 a text file named **readme.txt** that contains the text **linux** * (T203, 2 pts) **VM1:** In each folder **source** create a file named **code.sh** that contains just the signature for bash scripts and a single command - **pwd** | /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:** **VM2:** Create a copy of the **/important/animal-stories.txt** file as **/data/animals/lions.txt** file which contains only lines that contain the **lion** text no matter the register (size of the letters) or position in a sentence
* (T209, 2 pts) **VM2:** **VM2:** Create a copy of the **/important/animal-stories.txt** file as **/data/animals/colors.txt** file which contains only the **unique** in alphabetic order for all records about **tigers**

#### Users and Permissions - 8 tasks, 12 pts

* (T301, 2 pts) **VM1:** Create a user **john** with full name **John Smith**, with **some password** and auto-created **home folder**
* (T302, 2 pts) **VM1:** Create a user **jane** with full name **Jane Parker**, with **some password** and auto-created **home folder**
* (T303, 1 pts) **VM1:** Create a group named **team**
* (T304, 1 pts) **VM1:** Make both users, **john** and **jane**, part of the **team** group
* (T305, 1 pts) **VM1:** Make user **jane** and group **team** owners of the **/data/projects** folder and all its **sub-folders** and **files**
* (T306, 2 pts) **VM2:** Create a user **jim** with full name **Jim Beam**, with **some password**, auto-created **home folder**
* (T307, 2 pts) **VM2:** Create a group named **powerteam**
* (T308, 1 pts) **VM2:** Make the user **jim** part of the **powerteam** group

#### Software and Services - 7 tasks, 10 pts

* (T401, 1 pts) **VM1:** Visit **https://repos.zahariev.pro/** and follow the instructions to register the repository on the machine
* (T402, 2 pts) **VM1:** Create a file **/data/repos/packages.txt** with the list of all available packages in the repository registered in T401
* (T403, 1 pts) **VM1:** Install the **hello-lsa** package from the repository registered in T401
* (T404, 2 pts) **VM2:** Install **NGINX Web Server**, start it, and enable it to run on boot
* (T405, 1 pts) **VM2:** Install the **cowsay** package and execute it with the following command **cowsay 'Hello LSA(1)' > ~/cowsay.txt**
* (T406, 1 pts) **VM2:** Download the appropriate package for your distribution:
  + For ***Red Hat-based*** and ***openSUSE-based***, 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 ***Debian-based***, 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)
* (T407, 2 pts) **VM2:** Install the downloaded package (do not delete the downloaded file)

#### Scripting and Schedules - 7 tasks, 11 pts

* (T501, 1 pts) **VM1:** Create a bash script file *(should contain signature at least)* named **processes.sh** in the **/data/scripts** folder
* (T502, 1 pts) **VM1:** Change permissions of the **/data/scripts/processes.sh** file to **executable** for **everyone**
* (T503, 4 pts) **VM1:** When executed the **processes.sh** script should capture **the date and time and the number of running processes** and **store (append) the results** in the **/tmp/processes.log** file.

*For example, if the script is executed on* ***2023.10.24*** *at* ***10:15:53*** *and there are* ***82*** *processes, the row that would be stored in the file should be* ***2023-10-24 10-15-53 -> 82***

* (T504, 1 pts) **VM1:** **Schedule** the script for the **exam** user to execute **every five minutes**
* (T505, 1 pts) **VM2:** Create a bash script file *(should contain signature at least)* named **stars.sh** in the **/data/scripts** folder
* (T506, 1 pts) **VM2:** Change permissions of the **/data/scripts/stars.sh** file to **executable** for **everyone**
* (T507, 2 pts) **VM2:** When executed the **stars.sh** script should accept one parameter (if not given, should return an error and exit) and print the symbol **\*** in a row as many times as specified by the parameter.

*For example, if the script is executed like this* ***/data/scripts/stars.sh 5*** *then the result will be* ***\*\*\*\*\****