Linux Server Administration (First level full-time studies)

Mgr inz Patient ZIHISIRE MUKE
Department of Applied Informatics
Faculty of Information and Communication Technology
Wrocław University of Technology
Patient.zihisire@pwr.edu.pl

LAB 6:

Version 0.0-290322

Lab 6. Performing data compression. Backing up and recovering. Scheduling.

Prequisites:

- LSA lecture about archiving, data backup and scheduling (slides05.pdf)
- ebook available at http://linuxcommand.org/tlcl.php (Chapter 18 - Archiving and Backup)
- quota – online materials - PL(https://margib.blogspot.com/2012/10/quota-zarzadzanie-systemem-plikow-i.html), EN(https://linuxhint.com/disk_quota_ubuntu/)
- cron – online materials - EN(https://www.computerhope.com/unix/ucrontab.htm)

Exercises:

Task 1: quota (disk limits) (from the system administrator level) (2 points)

1. Taking into consideration that using quota on Linux may cause irreversible changes in operating system, it is strongly advised to backup virtual hard drive that is being used by your virtual machine (in VirtualBox it is a file with .vdi extension)
2. Create a new user account and rename with “YourReallyLastName” (example: “muke”). Using materials from lecture or any online materials install quota software on your machine. Set required options in *etc*/fstab file, to enable quota for your system partition (usually it is a partition with <mount points> set to “/” in */etc/fstab).* Restart your PC (virtual machine) and enable quota afterwards (quotacheck, quotaon)
   - check data limit set for “YourReallyLastName” created account, check how much of that limit is currently being used
   - set a small limit for “YourReallyLastName” created account, for example 30MB, with soft limit set to 20 MB
   - use a command ‘base64 /dev/urandom | head -c 1M > file.txt’ where 1M means 1MB and create several files so that “YourReallyLastName” created account exceeds soft limit. Check what happens when “YourReallyLastName” created account exceeds hard limit
   - in a similar way tests limits for maximum number of files that user is allowed to create (inodes setting in quota)
3. Do the exercise described above but this time set limits for a group of users. Add new users if necessary and prepare a dedicated group for them.

Task 2. Data compression and decompression: gzip, bzip, zip. And Creating tar archives (including compressed). (2 points)

4. In your home directory create a ‘compression+3last ID digits’ directory. Example:” compression123”. Inside create 4 text files with size equal to 1MB (you can use command from previous exercises). Create a new archive (tar) and add there all the files you have just created. What is the size of that archive? Why?
5. Add one of files created in previous exercise (or create a new one), archive and compress it using tar command with appropriate arguments (gzip compression).

Compress the same file using gzip command. Compare sizes of those files. Use 'file' command to check the type of compressed files.

6. List contents of the archive created in exercise 4. Create a new directory and extract all files there.
7. What types (relative or absolute) of paths did you observe in previous exercise? Create an archive again, but this time preserve full paths to archived files. Extract this archive and examine results.
8. In your home directory create a new archive with all files from /etc/systemd with .conf extension. After that extract this archive in your home directory. Check file ownership of extracted files and compare that with original files. Create another archive and extract it maintaining the original owner.
9. From the archive created in exercise 7 extract only one file of your choice.

NB: After the checking and marking of this is done by the lecturer, you can delete the created 'compression+3last ID digits' directory.

Task 3. Scheduling non-cyclic operations, with "at" commands ("at" with associated commands) and Scheduling cyclical operations, including /etc/crontab files and local user crontab files, and configuring local user cyclic tasks (crontab -e). (2 points)

10. Install (if not yet installed) 'at' program (sudo apt-get install at). Using 'at' schedule creation of a file named created_by_at in ~/at_testing directory. Schedule more than one task in similar way. Check the queue of scheduled tasks, remove one of them and check if it was properly removed. Schedule creation of a file called "humptydumpty.txt", which should be created at next Tuesday at 19.05 (7.05 PM). Schedule another task that will be executed after 10 minutes starting from the moment when you are scheduling it.
11. Using cron schedule following actions for current user (crontab -e). The actions will be executed:
    - every day at midnight
    - each friday at 5 PM
    - in march, june, september and december (quarterly)
    - every hour from monday to friday between 8 AM and 5 PM
    - every 5 minutes
    - every 2 hours
    - at every Sunday at 11 PM

Verify if all the tasks are properly defined, list all of them. Use the right command to remove one of previously defined exercises.
Using system directories (/etc/cron.{weekly,daily…}) as an administrator schedule a job that will be invoked at every hour.