Linux utilities that every developer should know

by Roman Pukhliy

Work with files	Move (rename) file	`mv /path/to/source /path/to/destination`
	Copy file to directory	`cp /path/to/source /path/to/destination_directory`
	Copy the directory with all the files inside it	`cp -r /path/to/source_directory /path/to/destination_directory`
	Copy several files	`cp /path/to/file1 /path/to/file2 /path/to/destination_directory`
	Delete a specific file	`rm /path/to/file`
	Delete files with extension *.txt*	`rm -rf *.txt`
	Delete directory with all files inside (flag *-f* for deletion without confirmation)	`rm -rf /path/to/directory`
	Change file owner	`chown user:group /path/to/file`
	Change the owner of the directory and all files inside it	`chown -R user:group /path/to/directory`
	Give all users the permission to read and write to the directory and all files inside it	`chmod -R a+rw /path/to/directory`
	Make the file permission the same as the other file	`chmod --reference=/path/to/source /path/to/destination`
Work with disk drives	Mount disk */dev/sdb1* with mount point */mnt/usb*	`mount /dev/sdb1 /mnt/usb`
	Mount device with file system *ext4* only for reading	`mount /dev/sdb1 /mnt/usb -t ext4 -o ro -o noload`
	Unmount mount point	`umount /mnt/usb`
	Force unmount file system	`umount -f /mnt/usb`
	Copy blocks of one device to another	`dd if=/path/to/input of=/path/to/output`
	Write the image to the device	`dd bs=4M if=/path/to/linux.iso of=/dev/sdx`
Reading logs	Display the first 20 lines from the file	`head -n 20 access.log`
	Display the last 30 lines of the file	`tail -n 30 error.log`
	Launch *tail* in the tracking mode of new line	`tail -f access.log`
	Open file for paginated output	`less access.log`
	Open file with line number displaying	`less -N access.log`
File system search	Find files with name *netdata.conf*	`find -name 'netdata.conf'`
	Find all files with extension *.conf*	`find / -name '*.conf'`
	Find all files with name *apache2*	`find / -type f -name 'apache2'`
	Find all directories with a name *nginx*	`find / -type d -name 'nginx'`
	Find all files larger than 100MB in your home directory	`find ~ -size +100M`
	Find in the home directory all files with a size less than 100MB	`find ~ -size -100M`
	Find all empty files in the home directory	`find ~ -empty`
	Delete all empty files in the home directory (*{}* replaced by the file name)	`find ~ -empty -exec rm -rf {} \;`
	Find word *Forbidden* in file *error.log*	`grep 'Forbidden' error.log`
	Find word *forbidden* in file *error.log* (case-insensitive search)	`grep -i 'forbidden' error.log`
	Display the number of matches found	`grep -c 'forbidden' error.log`
	Display an additional 2 lines after the match	`grep -i -A2 'forbidden' error.log`
	Display an additional 2 lines before the match	`grep -i -B2 'forbidden' error.log`
	Display an additional 2 lines before and after the match	`grep -i -C2 'forbidden' error.log`
	Find a phrase *access denied* in all files in the folder *~/.pm2/logs*	`grep -i -r 'access denied' ~/.pm2/logs`

Linux utilities that every developer should know (part 2)

by Roman Pukhliy

Dealing with processes	Display a list of all processes	`ps aux`
	Display only *node* processes	`ps aux | grep node`
	Display processes as a tree, show only *pid* and *command*	`ps -e -o pid,args –forest`
	Send signal *SIGTERM* (sent by default) to the process with *pid 8888*	`kill -SIGTERM 8888`
	Send *SIGKILL* (force terminate the process) to the process with *pid 8888*	`kill -9 8888`
	Stop all processes named *node*	`killall node`
	Display processes whose parent is process with *pid 3607*	`pgrep -P 3607`
	Display the processes that opened the file */etc/hosts*	`lsof /etc/hosts`
	Find the process which took port 80	`lsof -i :80`
Running processes in background	Run the process in the background	`ping google.com &`
	Run several processes in the background	`ping goole.com & nmap 192.168.1.* &`
	Display a list of background processes	`jobs -l`
	Get access to the process (put it into priority mode)	`%1`
	Bring the process back to background	`CTRL+Z` `%1 &`
	Start a new screen session	`screen`
	Transfer session to detached mode	`CTRL+A+D`
	Start the process in a new session in detached mode	`screen -d -m ping google.com`
	View the list of sessions	`screen -ls`
	Attach screen session	`screen -R [session id]`
Deal with the network	Run *HEAD* request (get headers only)	`curl -I http://google.com`
	Run *POST* request with data sending	`curl -d 'first_name=John&last_name=Doe' http://google.com`
	Send *JSON* to server	`curl -d '{"name":"John"}' -H 'content-type: application/json' http://google.com`
	Download file (similar to using *wget*)	`curl -O http://google.com/1.png`
	Track packets that were sent from the local machine	`tcpdump src 127.0.0.1`
	Track packets that came to the local machine through a specific network interface	`tcpdump dst 127.0.0.1 -i eth0`
	Display a list of network interfaces	`tcpdump --list-interfaces`
	Track packages that are gone from the local machine to *google.com*	`tcpdump src 127.0.0.1 and dst google.com -n`
	Display captured packets in *ASCII*	`tcpdump dst google.com and port 80 -A`
	Track packets to a specific *IP* and *port*	`tcpdump dst google.com and port 443 -n`
	Scan a specific server	`nmap -sP 217.160.0.201`
	Scan local network	`nmap -sP 192.168.1.*`
	Try to determine the server OS	`nmap -O 192.168.1.8`
	Scan server ports (use *-sV* to determine the version of the service)	`nmap -Pn 192.168.1.8`
	Run a quick scan (scanning of the common services)	`nmap -F 192.168.1.8`
	Scan specific ports (use *-open* to display only open ports)	`nmap -Pn -p 80,443 192.168.1.8`
	Scan port combination (*U* – UDP port, *T* – TCP port, *21-25* – port range)	`nmap -p U:53,111,137,T:21-25,80,139,8080 192.168.1.1`
	Scan 10 top ports (*ssh, ftp, http* ...)	`nmap -Pn --top-ports 10 192.168.1.8`

Linux utilities that every developer should know (part 3)

by Roman Pukhliy

System resources utilization	Launch interactive process monitor	`top` `htop`
	Run I/O monitor	`iotop`
	Display space usage	`du /path/to/directory`
	Analyze space usage	`ncdu /path/to/directory`
View system information	Display OS name and version	`lsb_release -a`
	See the full list of all devices	`lshw`
	See processor information	`lscpu`
	Display RAM Information	`free -h`
	See information about all mount points	`df -h`
	Display information about all available block devices	`lsblk`
Other utilities	Start the process with update every 500 ms	`watch -n0.5 ls -laS`
	Reset terminal	`reset`
	Display the calendar for the current year	`cal -j`
	Display calendar for June 2021	`cal -j 6 2021`
	Display information about the current user	`id`
	Display current user name	`whoami`
	Convert string to base64	`echo Hello | base64`
	Decode base64 string	`echo "SGVsbG8K" | base64 -d`
	Format json	`echo '{"status":1}' | jq` `curl https://opinionated-quotes-api.gigalixirapp.com/v1/quotes | jq`