# Basic Commands

|  |  |  |
| --- | --- | --- |
| **command** | **description** | **example** |
| cd | change directory | cd <directorya> cd (change to home directory) cd ~ (change to home directory) cd - (change to last directory) |
| cp | copy file(s) or directories |  |
| date | Display current system date and time |  |
| echo | display text | echo <text> echo $SHELL (display current shell name) |
| exit | end session |  |
| find | find files / directories matching the provided criteria | find recursively from the provided directory all files/directories where ...  last\_modification\_time is older than 5 days and list them in details:  find . -mtime +5 -exec ls -lt {} \;  name contains pattern 'log':  find . -name "\*log\*"  name begins with 'agn.' and type is directory and delete those recursively:  find /opt/tomcat/work -name "agn.\*" -type d -exec rm -Rf {} \;  find all files (type = file) in given directory which are older than 30 days and move them to directory logarchive  find /opt/tomcat/logs -mtime +30 -type f -exec mv {} ./logarchive \;  find files bigger than 1MB  find . -size +1000k -exec ls -la {} \; |
| ln | create link to a file or directory | ln -s <file> <linkname> (create softlink to <file>) |
| ls | list directory entries | ls -a (list also hidden files) ls -l (long (detailed) list of files) ls -alFh <dir> (list symbolic links) ls -ltr (sort by timestamps reversely) |
| man | manual page of a command | man ls (show man page section 1 of ls) man cron 8 (show man page section 8 of command cron) |
| mkdir | make directory |  |
| mv | move / rename of file / directory | mv <filea> <fileb> |
| pwd | print working directory |  |
| rm | remove file / directory | rm <filea> rm -Rf <directorya> (remove recursively the directory with all its files and subdirectories) |
| rmdir | remove empty directory, only works for empty directories, otherwise use rm -Rf directory | rmdir <emptydirectory> |
| sudo | "substitute user do" - execute command as root user | sudo su (become root) sudo su - (become root and execute root's start scripts) sudo su user (become user) sudo su - user (become user and execute user's start scripts) |
| touch | create an empty file in the current directory | touch <filename> (change file timestamps - defaults to now) |

## Text processing

|  |  |  |
| --- | --- | --- |
| **command** | **description** | **example** |
| grep | find a file content and print lines matching a pattern | find a string in all files of the current directory:  grep AGN0004574 \*  find a string and list only the filename  grep -l AGN0004574 \*  find a string and count only the hits per file  grep -c WorkflowProcessingException \*  find all lines which do NOT match the given pattern  grep -v WorkflowProcessingException \*  combined with a find command  find . -name "\*.log" -exec grep -l WorkflowProcessingException {} \;  find all occurrences of <string> in all given files showing 3 lines before and 50 lines after each occurrence:  grep -A 50 -B 3 <string> `ls -rt1 STATUS\*.xml`  grep continuously  tail -f <file>|grep --line-buffered <string> |
| head | print beginning of input | print first 5 lines:  head -n 5 |
| less | show file and move backwards or forwards | less <filename> |
| tail | print end of input | print last 5 lines:  tail -n 5  show newest file:  ls -ltr | tail -n 1 |
| wc | wordcount | count number of lines:  wc -l  count number of files in current directory:  ls | wc -l |
| xargs | build and execute command lines from standard input | find . -name '\*' -print0 | xargs -0 <command> |

## Disk space

|  |  |  |
| --- | --- | --- |
| **command** | **description** | **example** |
| df | report filesystem disk space usage | df -h (print report in "human readable" form)  df -i (print inodes) |
| du | estimate file space usage | du -h . |

## Processes / Networking

|  |  |  |
| --- | --- | --- |
| **command** | **description** | **example** |
| dig | DNS lookup utility |  |
| host | find domain name for ip | host 192.168.1.82 |
| ifconfig | show or change interface configuration |  |
| lsof | list open files | list all internet network files:  lsof -i  count all open files of tomcat:  lsof |grep tomcat|wc -l  more examples:  lsof -u agn | wc –l  lsof -u root | wc -l  lsof | wc -l |
| netstat | Print network connections, routing tables, interface statistics, masquerade connections, and multicast memberships | netstat -vatn |
| nslookup | query Internet name servers interactively | nslookup <server> |
| ping | test reachability of a machine | ping 192.168.6.83 |
| ps | list processes | ps axu (list processes of all users, with or without tty and print owners) |
| route | show / manipulate the IP routing table |  |
| traceroute | print the route packets take to network host |  |
| ulimit | report the resources available to the shell and to processes started by it e.g. stack size, number of open files, max user processes | list all system resource limitations:  ulimit -a  more examples:  ulimit -Hn  ulimit -Sn |

## Scheduling

|  |  |  |
| --- | --- | --- |
| **command** | **description** | **example** |
| crontab | Show / edit scheduler table of current user | crontab -l (list crontab) crontab -e (edit crontab) |

## System infos

|  |  |  |
| --- | --- | --- |
| **command** | **description** | **example** |
| file | determine file type | file <filename> file -bi <filename> (print brief mime type) |
| uname | print system information | uname -a (show full system information,  "Linux applicationsrv1 2.6.32-5-amd64 #1 SMP Mon Feb 25 00:26:11 UTC 2013 x86\_64 GNU/Linux") |
| whereis | searches a command in the path | whereis <command> |
| which | which command executable is used | which <command> |
| who | Show users on the system |  |
| whoami | Show your own user |  |

# Unsortiert

grep -A 50 -B 3 ECO2000011583 `ls -rt1 STATUS\*.xml`

Nach eingegangenen Dateien suchen, die nicht älter als zwei Tage sind:

ll -t `find . -ctime -2 -exec grep -l GenTT0000737452 {} \;`

wget --post-file="AUFTRAG.xml" --header="Content-type: text/xml" http://localhost:8193/vdforder/in

# Useradd

useradd mmeier

passwd mmeier

mkdir /home/mmeier

chown mmeier:mmeier /home/mmeier

nano /etc/passwd # Bash als Shell festlegen

nano /etc/sudoers # sudo-Rechte anlegen

mmeier ALL=NOPASSWD: ALL