

Linux

Grundlagenhandbuch

- [Common Commands](#)
- [SSH Setup & Security](#)
- [Cronjobs](#)

Common Commands

Help Commands

Command	Description
compgen -c	Show all commands
compgen -a	Show all alias commands
compgen -A function	Show available functions
man <command>	Show manual page of a command (more info than --help)
<command> --help	Show help page of a command (less info than man command)
<command> --version	Show version of a command

File & Folder Commands

Command	Description
ls -lah	List files/folders in current directory (+infos)
mkdir newdir	Create a new folder
cd somefolder	Change/switch to another directory
cd /mnt/	change/switch to another drive
cp file.txt file-copy.txt	Copy a file
mv file.txt file-renamed.txt	Rename a file
mv file.txt /tmp	Move file to another folder
rm file.txt	Remove a file
rm -rf folder	Remove entire folder (recursive)
ln -s original target	Create a symlink
pwd	Show path of current folder

<code>chown <arguments> <user>:<group> file.txt</code>	Change Owner and Group of file (use -R for rekursive)
--	---

Search & String Commands

Command	Description
<code>find . -name "file.txt"</code>	Search for a file (recursive)
<code>find . -name "*.txt"</code>	Search for all txt files
<code>grep -rin "search string"</code>	Search for a string in current folder (recursive)
<code>grep -rin "search string" cut -c 1-100</code>	Search for a string in current folder + limit output of string
<code>sed -Ei 's/someString/newString/g' file.txt</code>	Replace "someString" with "newString" in a file
<code>head file.txt</code>	Show first 10 lines of file
<code>tail file.txt</code>	Show last 10 lines of file
<code>cat file.txt</code>	Show full content of file
<code>cat file.txt grep "search string"</code>	Pipe output of "cat" command to grep + filter string

Terminal Search, Shortcuts & Autocompletion

Command	Description
Ctrl + a	Jump to beginning of line
Ctrl + e	Jump to end of line
Ctrl + Arrow Left	Jump word backward
Ctrl + Arrow Right	Jump word forward
Ctrl + w	Remove word backward
Alt + w	Remove word forward
Ctrl + p	Go through command line history backward
Ctrl + n	Go through command line history forward

Ctrl + r	Reverse Search in command history: <ul style="list-style-type: none"> - Type in something to search - Type Enter to execute the searched command - Type "Ctrl + r" to search further in reverse - Type "Ctrl + s" to search further in inverse - Tip: In only a few cases it makes sense to use the inverse search (Ctrl + s) from the beginning, but usually you want to search in reverse from the latest history commands
Ctrl + s	Same as "Ctrl + r" but inverse (starts from the beginning of the history)
Ctrl + f	Autocomplete a suggestion (Fish Shell only)

OS Helper Commands

Command	Description
du -csh .	Show size of current folder
df -h	Show partitions / drives
lsb_release -a	Show OS Version / Informations

Remote / Host Commands

Command	Description
wget https://www.somewhere495848.com/test.txt	Download a file
curl -LI https://www.google.com	Show Header Informations of URL
whois <host>	Show NS, DNS, Register about a host (e.g.: whois google.com)
host <host>	DNS Lookup of a host (e.g.: host google.com)
dig <host>	DNS Lookup of a host (verbose) (e.g.: dig google.com)
rsync -avz myfolder somehost:/var/www/	Sync a folder to a remote host and specific path
rsync -avz somehost:/var/www/ .	Sync from remote host and specific path to local folder
rsync -avz <dirSource> <dirTarget>	Sync entire Folder (locally)

<code>rsync -avz <dirSource>/ <dirTarget></code>	Sync only contents of folder (locally)
--	--

Command-Line Operators

Command	Description
<code><command> > file.txt</code>	Write output of a command into a file (creates/overwrites file)
<code><command> >> file.txt</code>	Write output of a command into a file (appends at end of file)
<code><command1> && <command2></code>	Do command2 only if command1 succeeds
<code><command1> <command2></code>	Do command2 only if command1 fails
<code><command1> <command2></code>	Pipe command1 output into command2
<code>!</code>	Negation Operator (e.g. <code>find . ! -user "root"</code>)
More about operators: linux-command-operators	

Other helpful Commands

Command	Description
<code>bash myCustomScript.sh</code>	Execute a bash script
<code>history tail</code>	Show last 10 lines of your history (commands)
<code>htop</code>	Very helpful utility to show all process, cpu, ram, ..
<code>ps aux grep http</code>	Show processes who accesses http
<code>which <command></code>	Show path location of a command
<code>tar -cvf myfolder.tar myfolder</code>	Create a compressed tar file of a folder
<code>tar -xvf myfolder.tar</code>	Extract a compressed tar file
<code>zip -r myfolder.zip myfolder</code>	Create a compressed zip file
<code>unzip myfolder.zip</code>	Extract a compressed zip file
<code>EDITOR=nano crontab -e</code>	Set Editor for current shell only (temporarily)
<code>select-editor</code>	Set Editor for user (usually permanently)

<code>vim file.txt</code>	Edit a file in the Vim Editor
<code>nano file.txt</code>	Edit a file in the Nano Editor
<code>sudo su - www-data -s /bin/bash</code>	Switch User to example "www-data"

SSH Setup & Security

SSH File Structure

File-/Folder Structure	Description	Security
~	User Home Directory (e.g. ~ can be /home/username)	
~/.ssh	SSH Ordner im Home Verzeichnis	<pre>chmod 700 ~/.ssh</pre>
~/.ssh/config	<p>Erstelle einen neuen Host Eintrag in der ssh config, der Aufbau sollte wie folgt aussehen:</p> <pre>Host SomeHostAliasName HostName domain.tld # or IP User root # or another user</pre> <p>Eine Verbindung kann dann wie folgt durchgeführt werden:</p> <pre>ssh SomeHostAliasName</pre>	<pre>chmod 600 ~/.ssh/config</pre>
~/.ssh/id_rsa	Dein Private Key (niemals an andere übermitteln!!)	<pre>chmod 600 ~/.ssh/id_rsa</pre>
~/.ssh/id_rsa.pub	Dein Public Key (zum übermitteln an Dritte für Remote-Server Einrichtung)	<pre>chmod 600 ~/.ssh/id_rsa.pub</pre>
~/.ssh/authorized_keys	Public Keys die Zugriff auf den aktuellen Host haben	<pre>chmod 600 ~/.ssh/authorized_keys</pre>
~/.ssh/known_hosts	Einträge zu (trusted) Hosts (Einträge werden i.d.R. automatisch ermittelt und per User Prompt zur Bestätigung erfragt)	<pre>chmod 600 ~/.ssh/known_hosts</pre>

SSH Commands

Command	Description
<code>ssh someHostAliasName</code>	Connect to a host from your <code>~/.ssh/config</code>
<code>ssh-keygen -t rsa -b 4096 -C "your_email@example.com"</code>	Create a 4096Bit encrypted SSH Key
<code>ssh-copy-id someHostAliasName</code>	Copies your current ssh pub key to a remote host (Alternatively connect to the ssh host and add your public key content to the file <code>~/.ssh/authorized_keys</code>)
<code>ssh -Tv git@github.com</code>	Analyze if a ssh connection is possible to a host (e.g. <code>git@github.com</code>)
<code>ssh -o PubkeyAuthentication=no -o PreferredAuthentications=password someHostAliasName</code>	Check if a host allows password authentication

SSH Hardening:

This hardens SSH + Disables Root Access (do this only if you know what you are doing!!)

Edit `sshd_config` (e.g. `vim /etc/ssh/sshd_config`) and change/add the following:

```
PermitRootLogin no          # or "prohibit-password"
PubkeyAuthentication yes
PasswordAuthentication no   # or use Match Blocks instead (see:
https://ostechnix.com/disable-ssh-password-authentication-for-specific-user-or-group/)
PermitEmptyPasswords no
ChallengeResponseAuthentication no
UsePAM no
X11Forwarding no

# Optional:
# Match User app-*
#     PasswordAuthentication yes
#
# Match User admin-*
#     PasswordAuthentication no
```

After changing the file restart the ssh service: `systemctl restart sshd` and verify if you are still able to connect (use a different user than root)!

Cronjobs

Default PHP-Script Aufruf:

```
*/5 * * * * [path/to/php] [absPath/to/script]
```

PHP-Script Aufruf mit env Variable:

e. g. `**** TYPO3_CONTEXT=Development/Server /[absPathToPHP]/php7.4-cli -f
~/[absPath]/typo3/sysext/core/bin/typo3 scheduler:run`

```
*/5 * * * * [ENV-Variable] [path/to/php] [absPath/to/script]
```