

# Linux - Bag of Tricks

## Introduction

This document has many useful command.

## Linux Set Time Examples

You can also simplify format using following syntax:

```
date +%Y%m%d -s "20081128"
```

To set time use the following syntax:

```
date +%T -s "10:13:13"
```

Use the following syntax to set new data and time:

```
date --set="STRING"
```

For example, set new data to 2 Oct 2006 18:00:00, type the following command as root user:

```
date -s "2 OCT 2006 18:00:00"
```

OR

```
date --set="2 OCT 2006 18:00:00"
```

## Rsync Copy Examples

This is to move files from one server to another

Ending the folder WITHOUT a "/" slash means copy that folder everything in that folder

Ending the folder WITH a "/" slash means copy everything within that folder

Example for "remote to local" location

```
rsync -chavzP --stats --progress -e ssh user@remote_host:/remote_folder/dir1/ /local_folder/dir1/
```

Example for "local to remote" location

```
rsync -chavzP --stats --progress -e ssh /local_folder/dir1/ user@remote_host:/remote_folder/dir1/
```

## Rsync Auto Login while sending

Example to add a Rsync key on the remote server

On the local server simply login as a given user ex: ROOT or USER

```
ssh-keygen -t rsa
```

If it already exists simply hit "n"

```
Generating public/private rsa key pair.  
Enter file in which to save the key (/root/.ssh/id_rsa):  
/root/.ssh/id_rsa already exists.  
Overwrite (y/n)?
```

If not then simply hit enter through all options

Example: of using ROOT

```

Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:JoMN/cxvsqZWBHws4eyrU5Q0F0qRe//44qdrriQmbU root@DSS-US-TMAP-XXX
The key's randomart image is:
+---[RSA 3072]----+
|  .+=..      |
|  =B.+       |
|  ..=O       |
|  ==+o       |
|  = E=...    |
|  o... oo    |
|  ..o..++ o  |
|  .oo+o==*.  |
+----[SHA256]-----+
You have mail in /var/spool/mail/root

```

Run the following to add the key to the remote server, you can also use IP instead of host name

```
ssh-copy-id -i ~/.ssh/id_rsa.pub remuser@sfl-lin-001
```

Example of using a USER you will have to enter yes and the USER password

```

/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host 'sfl-lin-020 (192.168.136.80)' can't be established.
ED25519 key fingerprint is SHA256:oZnvrgY+2Xpd2/huaffvzLMBAgI52AMPUmq/LPLIXbE.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
remuser@dss-us-map-020's password:
tput: No value for $TERM and no -T specified
tput: No value for $TERM and no -T specified
tput: No value for $TERM and no -T specified
tput: No value for $TERM and no -T specified
tput: No value for $TERM and no -T specified
tput: No value for $TERM and no -T specified
tput: No value for $TERM and no -T specified
Number of key(s) added: 1
Now try logging into the machine, with: "ssh 'remuser@sfl-lin-020'"
and check to make sure that only the key(s) you wanted were added.

```

**Optional:** If the command cannot be run above you can copy the key to the remote server manually into the “authorized\_keys” file

```

cd
cd .ssh
vi authorized_keys

```

**Optional:** Change the permissions on the local server

```
chmod 600 ~/.ssh/*  
chmod 711 ~/.ssh  
chmod 711 ~
```

## Synology Rsync

```
rsync -aXHmS --syno-acl /volum1/[xxx] /volume2/[xxx]
```

-a, --archive archive mode; equals -rlptgoD (no -H,-A,-X)  
-p, --perms preserve permissions  
-X, --xattrs preserve extended attributes  
-o, --owner preserve owner (super-user only)  
-g, --group preserve group  
--syno-acl copy Synology ACL data

I use the following options myself:

```
rsync -avhxWog --stats --backup --suffix $OLDSUFFIX --exclude-from=$RSYEXCL --syno-pseudo-root
```

No idea why I list options "og" since they're implied by -a, but it works...

Regards, Arild

PS: "rsync --help" lists all available options for rsync

## Find and Replace String with `sed`

There are several versions of `sed`, with some functional differences between them. macOS uses the BSD version, while most Linux distributions come with GNU `sed` pre-installed by default. We'll use the GNU version.

The general form of searching and replacing text using `sed` takes the following form:

```
sed -i 's/SEARCH_REGEX/REPLACEMENT/g' INPUTFILE
```

Cop

- `-i` - By default, `sed` writes its output to the standard output. This option tells `sed` to edit files in place. If an extension is supplied (ex `-i.bak`), a backup of the original file is created.
- `s` - The substitute command, probably the most used command in `sed`.

- `///` - Delimiter character. It can be any character but usually the slash (`/`) character is used.
- `SEARCH_REGEX` - Normal string or a regular expression to search for.
- `REPLACEMENT` - The replacement string.
- `g` - Global replacement flag. By default, `sed` reads the file line by line and changes only the first occurrence of the `SEARCH_REGEX` on a line. When the replacement flag is provided, all occurrences are replaced.
- `INPUTFILE` - The name of the file on which you want to run the command.

## Find and Replace String with `sed` within `vi`

This is to search and replace a file globally withing vi

```
:%s/search_string/replacement_string/g
```

## Kill Users in Linux

This is to be used when trying to kill users using the connection, replace the `?` with the number of the session.

```
pkill -KILL -t pts/?
```

## Create a CERT

First, you need to generate the private key and the Certificate Signing Request (CSR). You can do this via the `openssl` command:

```
openssl req -nodes -newkey rsa:2048 -keyout privatekey.key -out mail.csr
```

Then, generate a signing request

```
openssl x509 -req -days 365 -in mail.csr -signkey privatekey.key -out secure.crt
```

Create a localhost cert on the server

```
openssl req -newkey rsa:2048 -nodes -keyout /etc/pki/tls/private/localhost.key -x509 -days 365 -out  
/etc/pki/tls/certs/localhost.crt
```

## Mariadb Log Rotate

If log file is large, try if the logrotate

```
logrotate --force /etc/logrotate.d/mariadb
```

## MySQL Fail to Start

If MySQL does not restart, it probably will not as the index of the log files will not be changed

```
:d /var/lib/mysql  
mv ib_logfile0 ib_logfile0.old  
mv ib_logfile1 ib_logfile1.old  
systemctl restart mariadb
```

## Configure Rsync

Useful for system migrations

Create a “/etc/rsyncd.conf” containing:

```
[root]  
exclude = /dev /etc/fstab /proc /sys  
path = /  
read only = yes  
list = yes  
uid = root  
gid = root
```

Enable and start:

```
systemctl enable rsyncd.service  
systemctl start rsyncd.service
```

## Change Run level

```
systemctl set-default multi-user.target
```

To switch from graphical to multi-user:

```
systemctl isolate multi-user.target;
```

## Change Local settings

```
# timedatectl set-timezone Europe/London
```

```
# localectl set-locale LANG=en_GB.UTF-8
```

```
# localectl set-keymap uk
```

Temporary change

```
$ loadkeys us
```

## Configure Alternate Authentication

```
authconfig-tui
```

## SSD Considerations

Change the value of “`issue_discards`” option from 0 to 1 in “`/etc/lvm/lvm.conf`”

```
#  
systemctl enable fstrim.timer
```

Adjust “`/etc/fstab`”

```
/dev/mapper/xxx /XXX    xfs    defaults,noatime,discard    0 0
```

Optionally set `/tmp` in RAM

```
# systemctl enable tmp.mount
```

## Adding a Disk

```
# parted /dev/sdx
```

```
mklabel gpt
```

```
unit s
```

```
mkpart primary 2048s 100%
```

```
set 1 lvm on
```

```
quit
```

```
# pvcreate /dev/sdx1
# vgcreate rl_ssd /dev/sdx1
# lvcreate -L 50GB -n mysql rl_ssd
# mkfs.xfs /dev/rl-ssd/mysql
# blkid /dev/sdc

# chown mysql:mysql /var/lib/mysql
```

## Growing a lvm partition

```
# parted /dev/sdc
```

```
(parted) unit b
```

```
(parted) print free
```

Number	Start	End	Size	Type	File system	Flags
1	31744B	5368709119B	5368677376B	primary		
	5368709120B	21474836479B	16106127360B			Free Space

```
(parted) resizepart 1 21474836479B
```

```
(parted) quit
```

```
# pvresize /dev/sdc1
```

## Updating Bootloader configuration

```
/etc/default/grub
```

```
grub2-mkconfig -o /boot/grub2/grub.cfg
```

NMAP Scan for all Open Ports



## TCP

```
sudo nmap -sT -p- onling.com
```

## UDP

```
sudo nmap -sU -p- onling.com
```

---

Revision #9

Created 2 September 2024 17:14:03 by Steve Ling

Updated 6 October 2024 20:45:08 by Steve Ling