First things to do

First steps after installing a new server to make sure nobody can capture it and use it in a way it was not intended. Make sure you work as fast and correct as possible until you reach *BREAKTIME*. That should not consume too much time and then you can think about what you want to install afterwards.

- https://www.thomas-krenn.com/de/wiki/Absicherung_eines_Debian_Servers

New user

Create new User

```
adduser sammy
```

Give sudo-rights

```
usermod -aG sudo sammy
```

Generate SSH key

```
ssh-keygen
```

Copy the public key to server

```
ssh-copy-id sammy@your_server_ip
```

test login

```
ssh sammy@serverip -p PORT
```

Configure SSH

Disable root login

```
sudo nano /etc/ssh/sshd_config
```

```
PermitRootLogin no
```

Disable password login

```
sudo nano /etc/ssh/sshd_config
```

```
ChallengeResponseAuthentication no
```

change SSH Port

It's more security by obscurity and not actually needed. It would reduce the amount of automated scans that reacht you ssh-port but is not really something to secure the server.

restart sshd

```
sudo systemctl restart sshd
```

fail2ban

```
sudo apt install fail2ban
```

```
sudo cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local
```

and configure (1 short-lock for 24 hours, one for 1 week block)

- https://www.booleanworld.com/protecting-ssh-fail2ban/
- https://blog.shanock.com/fail2ban-increased-ban-times-for-repeat-offenders/

```
sudo nano /etc/fail2ban/jail.local
```

```
#
# JAILS
#

#
# SSH servers
#

[sshd]

# To use more aggressive sshd modes set filter parameter "mode" in
jail.local:
# normal (default), ddos, extra or aggressive (combines all).
# See "tests/files/logs/sshd" or "filter.d/sshd.conf" for usage example and
details.
mode    = normal
port    = ssh
logpath = %(sshd_log)s
backend = %(sshd_backend)s
# input by stefan
# one day
findtime = 5400 ;1.5 hours
```

```
maxretry = 5
bantime = 86400 ;1 day

# input by stefan, longterm ban
# 30 attempts over 3 days result in a 1 week ban
[sshlongterm2]
port      = ssh
logpath   = %(sshd_log)s
banaction = iptables-multiport
findtime  = 259200 ;3 days
maxretry  = 10
bantime   = 604800 ;1 week
enabled   = true
filter    = sshd

[sshlongterm3]
enabled = true
filter = sshd
findtime = 15552000 ;6 months
maxretry = 15
bantime = 2592000 ;1 month
logpath   = %(sshd_log)s
banaction = iptables-multiport
```

Kontrolle mit tail

```
tail -10f /var/log/fail2ban.log
```

update

```
sudo apt update && sudo apt upgrade
```

UFW

1. https://linuxconfig.org/how-to-deny-all-incoming-ports-except-http-port-80-and-https-port-443-on-ubuntu-18-04-bionic-beaver-linux

Install und enable UFW and allow only SSH default [or Enable UFW and disable all inbound traffic from eth0 on all ports except SSH from my local IP (temporary, eventually I allow SSH globally due to potential for IP changes) and disable all outbound traffic except for port 80. (Because paranoia)] and for hosted websites port 80 and if you intend to use letsencrypt or something else port 443 too.

```
ufw allow APPLICATION
ufw enable
ufw disable
ufw status
```

Wichtige erste Ports zur Benutzung

```
ufw allow ssh
ufw allow http
ufw allow https
```

BREAKTIME

Pause, drink a cup of coffee, think about what you are going to do next and plan a little bit. The server now have some basic security.

Tools

Install tools you want(vim, tmux, htop, nmap, sysstat, net-tools)

Mailserver

Install and configure mailserver (postfix mit s-nail?) for automated messages from unattended upgrades or from other services.

Unattended Upgrades

Automatically just updates security-relevant updates. Can also update all updates, if you want. Can also send autmated messages if a mailserver is installed. A sugestion is to send on every update at first and change the setting later to “justOnError” in /etc/apt/apt.conf.d/50unattended-upgrades (multiple recipients separated with a komma)

Logrotate

Configure logrotate to rotate with dates instead of rolling numbers (easier for archive/backup)
https://linoxide.com/linux-how-to/setup-log-rotation-logrotate-ubuntu/

Logwatch

daily mail set up

Time-Related

Configure time-related stuff (tzdata, install ntp, setting the time zone to UTC)

Disable unrequired services

Disable any and all services that are not required for the purpose of the box, bind others to localhost, unless they need to listen on public interfaces. This reduces attack vectors.

Check this

- chef bootstrap (?)
- zsh (instead of bash), glances, rsync,
- Install most of the debug tools I've used in my life, just in case (lsof, gdb, iotop, slurm, strace)
- Enable Byobu - https://www.digitalocean.com/community/tutorials/how-to-install-and-use-byobu-for-terminal-management-on-ubuntu-16-04
- install etckeeper (etckeeper init, etckeeper commit -m initial)
- Webserver: letsencrypt
- vnstat
- install linuxbrew
- install git
- checkout my dot files from git
- install sudo and sudo-pam-auth. Configure it to work wi h ssh keys
- Learn Ansible? (install python-minimal for ansible)