

QUICK NAVIGATION

1. Basic install
2. Partitioning
3. Installing packages onto the system
4. Continuing the install from the system
5. Post install
 1. Optimization before installing apps
 2. Installing apps of choice
 3. Enabling services
 4. Kernel
 1. Tweak kernel settings
 2. Using different kernels
 5. Modify GRUB settings
 6. Customizing .bashrc
 7. Comfort zone & setting the look
 8. Custom scripts
 9. System maintaining
 10. Troubleshooting
6. Useful commands
7. Unsafe commands
8. Gaming
9. Security
10. Saving & reloading the system
11. Extra productiveness

Before updating or upgrading the system, check [the arch news](#) (could save your system)!

BASIC INSTALL

Setting keymap

```
loadkeys hu
```

Enable and connect to wifi (Optional)

```
rfkill unblock wifi
iwctl
device list
station **device** scan
station **device** get-networks
station **device** connect **SSID**
```

Sync time

```
pacman -Sy
timedatectl set-ntp true
```

Set fastest mirrorlist

```
pacman -Sy
pacman -S reflector
reflector -c Hungary -a 10 --sort rate --save /etc/pacman.d/mirrorlist
pacman -Sy
```

PARTITIONING

Check if running UEFI or BIOS

```
ls /sys/firmware/efi/efivars
```

If writes what's inside the folder: UEFI

If folder doesn't exist: BIOS

UEFI INSTALL

Creating partitions

```
fdisk /dev/sda
```

EFI g → n → ENTER → ENTER → +512M → t → 1
ROOT n → ENTER → ENTER → ENTER → w

Formatting the created partitions

```
mkfs.fat -F32 /dev/sda1  
mkfs.ext4 /dev/sda2
```

Mounting formatted partitions

```
mount /dev/sda2 /mnt  
mkdir /mnt/boot  
mount /dev/sda1 /mnt/boot
```

BIOS INSTALL

Creating partition

```
fdisk /dev/sda
```

n → ENTER → ENTER → ENTER → ENTER → w

Formatting the created partitions

```
mkfs.ext4 /dev/sda1
```

Mounting formatted partitions

```
mount /dev/sda1 /mnt
```

INSTALLING PACKAGES ONTO THE SYSTEM

[Basic packages](#)

```
pacstrap /mnt base linux linux-firmware
```

[Fstab](#)

```
genfstab -U /mnt >> /mnt/etc/fstab
```

[Configuring the system](#)

```
arch-chroot /mnt
```

[Time zone](#)

```
ln -sf /usr/share/zoneinfo/Europe/Budapest /etc/localetime
hwclock --systohc
```

[Localization](#)

```
nano /etc/locale.gen
uncomment en_US.UTF-8 UTF-8
      ↓
sed -i '177s/^#//' /etc/locale.gen

locale-gen
nano /etc/locale.conf
LANG=en_US.UTF-8
nano /etc/vconsole.conf
KEYMAP=hu
localectl set-keymap --no-convert hu
```

[Setting the system hostname](#)

```
nano /etc/hostname
arch

nano /etc/hosts
127.0.0.1    localhost
::1         localhost
127.0.1.1    arch.localdomain    arch
```

[Setting the root password](#)

```
passwd
```

Installing packages

```
pacman -S grub efibootmgr networkmanager network-manager-applet
wireless_tools wpa_supplicant dialog os-prober base-devel linux-headers reflector
git bluez bluez-utils xdg-utils xdg-user-dirs
```

Enable SSH connection (optional)

```
pacman -S openssh sshpass openssl
systemctl enable sshd
```

[Setting up GRUB \(UEFI\)](#)

```
grub-install --target=x86_64-efi --efi-directory=/boot --bootloader-id=GRUB
grub-mkconfig -o /boot/grub/grub.cfg
```

[Setting up GRUB \(BIOS\)](#)

```
grub-install --target=i386-pc /dev/sda
grub-mkconfig -o /boot/grub/grub.cfg
```

Enabling internet

```
systemctl enable NetworkManager
```

Enabling bluetooth (optional)

```
systemctl enable bluetooth
```

[Adding a new user](#)

```
useradd -mG wheel hakszi
passwd hakszi

EDITOR=nano visudo, and uncomment:
%wheel ALL=(ALL:ALL) ALL
      ↓
sudo sed -i '/%wheel ALL=(ALL:ALL) ALL/s/^#//g' /etc/sudoers
```

Finish initial setup

```
exit
umount -a
reboot
```

CONTINUING THE INSTALL FROM THE SYSTEM

(unplug USB)

Setting up wifi (optional)

```
sudo nmtui
```

Installing graphics drivers

[For AMD](#)

```
sudo pacman -S xf86-video-amdgpu
```

[For NVIDIA \(open source\) \(recommended\)](#)

```
sudo pacman -S xf86-video-nouveau
```

[NVS 310, GT & GTX 3xx 4xx 5xx series \(proprietary\)](#)

```
yay -S nvidia-340xx
```

[GTX 6xx 7xx 9xx series \(proprietary\)](#)

```
yay -S nvidia-390xx
```

[RTX 2060, GTX 10xx series \(proprietary\)](#)

```
yay -S nvidia-418xx
```

Setting up an AUR

[Install display driver](#)

```
sudo pacman -S xorg xorg-drivers
```

[Install desktop enviroment](#) [\(of personal choice\)](#)

XFCE

```
sudo pacman -Syu
sudo pacman -S xfce4 xfce4-goodies
```

↕

Cinnamon

```
sudo pacman -Syu
sudo pacman -S cinnamon gnome-terminal nemo-fileroller
```

#Note that the firewall, no-disturb toggle and auto-login script will only work with XFCE
See here: Firewall, no disturb mode and auto-login quick settings

[Installing login manager](#)

```
sudo pacman -S lightdm lightdm-gtk-greeter lightdm-gtk-greeter-settings
sudo systemctl enable lightdm
```

Set extra keyboard settings

```
localectl set-x11-keymap hu $USER
reboot
```

OPTIMALIZATION BEFORE INSTALLING APPS

Mount NTFS hard disk

```
su
pacman -S ntfs-3g
ntfs-3g /dev/your_NTFS_partition /mount/point
```

[Enable 32-bit applications \(recommended\)](#)

```
sudo sed -i '93s/^#//' /etc/pacman.conf
sudo sed -i '94s/^#//' /etc/pacman.conf
pacman -Sy
```

Check to make sure *[multilib]* and *include = /etc/pacman.d/mirrorlist* are uncommented

```
cat /etc/pacman.conf
```

[Optimize package making \(caching, using all cores, compression, etc...\)](#)

```
su
cd /etc
rm -rf makepkg.conf
wget https://pastebin.com/raw/9KVnvpbQ
mv 9KVnvpbQ makepkg.conf
*Note: Comment BUILDDIR=/tmp/makepkg on older computers with <8 GB of ram
```

Restore the original makepkg.conf file:

```
su
cd /etc
rm -rf makepkg.conf
wget https://pastebin.com/raw/RYEUK9ND
mv RYEUK9ND makepkg.conf
```

Enable parallel downloads (recommended)

```
sudo sed -i '/ParallelDownloads/s/^#//g' /etc/pacman.conf
```

Setting up an AUR manager

```
git clone https://aur.archlinux.org/yay-git
cd yay-git
makepkg -si
cd ~
sudo rm -rf yay-git
```

Setting up SNAP

```
yay -S snapd
sudo systemctl enable --now snapd.socket
sudo ln -s /var/lib/snapd/snap/snap
```

Installing GUI package manager:

```
yay -S octopi
```

Clean system & update & upgrade & reboot before installing apps

```
sudo pacman -Syu --noconfirm -q
sudo pacman -Syyu --noconfirm -q
paccache -rvk1
yay -Syu --noconfirm
yay --clean -Sc --noconfirm
sudo snap refresh
rm -rf /tmp/*
reboot
```

INSTALLING APPS OF CHOICE

Essentials:

```
sudo pacman -S pavucontrol alsa-firmware alsa-utils alsa-plugins pulseaudio-alsa
pulseaudio libcanberra a52dec dvdauthor dvgrab faac faad2 gst-libav libdca
libdvdcss libdvdnv libmad libmpeg2 exfat-utils fuse-exfat wget ifplugd python tar
rsync zip unzip fdupes bat curl youtube-dl bc htop pciutils xfce4-whiskermenu-
plugin xf86-input-synaptics viewnior ccache neofetch rust

yay -S update-grub timeshift zramswap xfce4-mixer etcher-bin woeusb-ng safe-rm
mkinitcpio-numlock
```

Most used apps:

```
sudo pacman -S virtualbox virtualbox-sdk virtualbox-host-dkms virtualbox-guest-iso
kdeconnect redshift libreoffice gnome-disk-utility gparted clamav krita qbittorrent
vlc ufw ufw-extras okular firefox engrampa torbrowser-launcher putty obs-studio
discord

yay -S rider podman docker dotnet-sdk-5.0-bin spotify qdirstat electronmail github-
desktop-bin pyakm stacer
```

```
snap installl teams-for-linux jpg2pdf pdfmixtool
```

Laptop-only:

```
yay -S slimbookbattery auto-cpufreq
```

For gaming:

```
yay -S dxvk-bin mesa vulkan-radeon mesa-libgl ttf-ms-fonts

pacman -S dotnet-runtime jre-openjdk lib32-alsa-plugins lib32-dbus lib32-freetype2
lib32-gnutls lib32-libgcrypt lib32-libgpg-error lib32-libldap lib32-libxml2 lib32-
mesa lib32-sdl2 lib32-vulkan-radeon libgcrypt lutris steam vulkan-icd-loader wine
wine-gecko wine-mono winetricks
```

Wallpapers

```
curl https://pastebin.com/raw/0EScRV8M
```

Themes

```
yay -S papirus-maia-icon-theme-git matcha-gtk-theme cantarell-static-fonts
capitaine-cursors

pacman -S noto-fonts-emoji adobe-source-code-pro-fonts adobe-source-han-sans-
hk-fonts adobe-source-han-sans-jp-fonts adobe-source-han-sans-kr-fonts adobe-
source-han-sans-otc-fonts adobe-source-han-sans-tw-fonts adobe-source-han-sans-cn-
fonts ttf-carlito ttf-droid gsfonts ttf-liberation xorg-fonts-misc
```

Theme:	Matcha-dark-aliz [GTK2/3]
Cursor theme:	Capitaine Cursors
Icons:	Papirus-Adapta-Maia [GTK2]
Font:	Cantarell Bold 10
Terminal font:	Source Code Pro Medium 14
See downloaded looks here: Themes	

Other apps:

<i>ncdu</i>	-	<i>Disk usage analyzer</i>
<i>lyx-git</i>	-	<i>LaTeX editor</i>
<i>vim neovim</i>	-	<i>Nano alternative for text editing</i>
<i>pkgbuild-watch</i>	-	<i>Monitor upstream for updates</i>
<i>firejail firetools</i>	-	<i>Linux namespaces sandbox program & GUI</i>
<i>calibre</i>	-	<i>Ebook management application</i>
<i>transmission-makepkg</i>	-	<i>makepkg download agent for magnet URIs</i>
<i>aur-out-of-date</i>	-	<i>Determines out-of-date AUR packages</i>
<i>net-tools</i>	-	<i>Configuration tools for Linux networking</i>
<i>nmap</i>	-	<i>Utility for network discovery and security auditing</i>
<i>ntop</i>	-	<i>A network traffic probe that shows the network usage.</i>
<i>archlinux-contrib</i>	-	<i>useful scripts</i>
<i>pacman-contrib</i>	-	<i>useful scripts</i>

ENABLING SERVICES

Start and enable RAM swap at boot

```
sudo systemctl enable zramswap.service
sudo systemctl start zramswap.service
```

Start and enable CPU frequency modifying

```
sudo systemctl enable auto-cpufreq.service
sudo systemctl start auto-cpufreq.service
```

Enabling TLP

```
sudo systemctl enable tlp.service
sudo systemctl start tlp.service
```

Enabling Trimming

```
sudo systemctl enable fstrim.timer
sudo systemctl start fstrim.timer
```


TWEAK KERNEL SETTINGS

#paste into /etc/sysctl.conf

```
net.ipv6.conf.all.disable_ipv6 = 1
vm.swappiness = 80
vm.vfs_cache_pressure = 300
vm.dirty_background_ratio = 5
vm.dirty_ratio = 10
vm.dirty_background_bytes = 4194304
vm.dirty_bytes = 4194304
vm.min_free_kbytes = 8192
kernel.sysrq = 1

#Network tweaks
net.ipv4.tcp_slow_start_after_idle = 0
net.core.default_qdisc = cake
net.ipv4.tcp_congestion_control = bbr
#Receive queue
net.core.netdev_max_backlog = 16384
#Max connections
net.core.somaxconn = 16384
#Max memory
net.core.rmem_default = 1048576
net.core.rmem_max = 16777216
net.core.wmem_default = 1048576
net.core.wmem_max = 16777216
net.core.optmem_max = 65536
net.ipv4.tcp_rmem = 4096 1048576 2097152
net.ipv4.tcp_wmem = 4096 65536 16777216
#UDP Limit
net.ipv4.udp_rmem_min = 8192
net.ipv4.udp_wmem_min = 8192
#TCP FastOpen feature
net.ipv4.tcp_fastopen = 1
```

USING DIFFERENT KERNELS

[Comparing hardened, original, LTS, and zen kernels](#)

[Harden kernel \(more stable\)](#)

```
su
pacman -S linux-hardened linux-hardened-headers
mkinitcpio -p linux-hardened
grub-mkconfig -o /boot/grub/grub.cfg
reboot
```

[Zen kernel \(for performance\)](#)

```
su
sudo pacman -S linux-zen linux-zen-headers
mkinitcpio -p linux-zen
grub-mkconfig -o /boot/grub/grub.cfg
reboot
```

[LTS kernel \(for long term support\)](#)

```
su
sudo pacman -S linux-lts
mkinitcpio -p linux-zen
grub-mkconfig -o /boot/grub/grub.cfg
reboot
```

[Remember last boot entry](#)

```
nano /etc/default/grub
modify:
    GRUB_DEFAULT=saved
add new line:
    GRUB_SAVEDEFAULT=true
```

MODIFY GRUB SETTINGS

Using Octopi download “grub-customizer”

Download distro theme of choice

<https://www.gnome-look.org/p/1482847>

Open grub customizer

Go into General settings

Set Boot default entry to 1 second

Set kernel parameters to:

quiet apparmor=1 security=apparmor udev.log_priority=3

Go into Appearance settings

Set theme to arch

CUSTOMIZING .BASHRC

```
#PS1="\[\033[38;5;1m\]\u\[$(tput sgr0)\] \[$(tput bold)\]\W\[$(tput sgr0)\] \\\$ \[$(tput sgr0)\]"
PS1="\[\033[38;5;51m\]\u\[$(tput sgr0)\] \W \\\$ \[$(tput sgr0)\]"
```

```
neofetch --ascii_distro archmerge --color_blocks off --cpu_temp C --refresh_rate on --os_arch on --
package_managers on --cpu_brand on --cpu_cores on --cpu_speed on --cpu_temp C --speed_type
current
#neofetch --ascii_distro blackarch --color_blocks off --cpu_temp C --refresh_rate on --os_arch on --
package_managers on --cpu_brand on --cpu_cores on --cpu_speed on --cpu_temp C --speed_type
current
#Actual used aliases
alias calculator='bc -l'
alias yaystat='sudo yay -Ps'
alias pacman='sudo pacman'
alias boot='systemd-analyze'
alias error='sudo journalctl -p 3 -xb'
alias firewall='sudo ufw status verbose'
alias bashrc="nano ~/.bashrc && source ~/.bashrc"
alias disk='ncdu'
```

```
#More readable / nicer look / better usability
alias vi='nvim'
alias vim='nvim'
alias mv='mv -vv'
alias ping='pretyping'
alias ls='ls --color=auto'
alias df='df -h'
alias diff='diff --color=auto'
alias free='free -m'
alias grep='grep --color=auto'
alias ip='ip -color=auto'
alias ls='ls --color=auto'
alias more='less'
alias nano='sudo nano'
alias lsblk='lsblk -e 7'
alias cat='bat'
```

```
#Are you sure?
#copy
alias cp='cp -i'
#move / rename
alias mv='mv -i'
#delete
alias rm='rm -i'
untar='tar zxf'
```

```
#Firewall, no disturb mode and autologin quick settings:
alias home='bash ~/.config/home.sh'
alias work='bash ~/.config/work.sh'
```

COMFORT ZONE & SETTING THE LOOK

Enable auto-login:

```
su
cd /etc/lightdm/
rm -rf lightdm.conf
wget https://pastebin.com/raw/R3UqReib
mv R3UqReib lightdm.conf
```

Revert to original lightdm config file:

```
su
cd /etc/lightdm/
rm -rf lightdm.conf
wget https://pastebin.com/raw/n8THjn0u
mv n8THjn0u lightdm.conf
```

Also for autologin:

```
sudo groupadd -r autologin
sudo gpasswd -a $USER autologin
```

Improve look of fonts

[Improve look of fonts](#)

[Stop beep sound:](#)

```
sudo mkdir /etc/modeprobe.d
sudo touch /etc/modeprobe.d/pcbeep.conf
nano /etc/modeprobe.d/pcbeep.conf
    blacklist pcspkr
reboot
```

Session and Startup → Application autostart → + → Add these:

```
xset b off
xset b 0 0 0
```

Disable F1 for help and F11 for fullscreen for terminal

```
nano ~/.config/xfce4/terminal/acceIs.scm
add these lines:
(gtk_accel_path "<Actions>/terminal-window/fullscreen" "F11")
(gtk_accel_path "<Actions>/terminal-window/contents" "F1")
```

[Set keyboard shortcut to launch whisker menu](#)

[One wallpaper across multihead](#)

[Set all java apps to use GTK theme settings](#)

[Activate numlock on startup](#)

[Dualboot arch with windows](#)

CUSTOM SCRIPTS

Firewall, no disturb mode and auto-login quick settings
curl https://pastebin.com/raw/Vna6dNRz
Alias them on page 12

Custom neofetch

```
rm -rf ~/.config/neofetch/config.conf
cd ~/.config/neofetch/
wget https://pastebin.com/raw/dWnPDBaX
mv dWnPDBaX config.conf
```

Custom polkit

```
su
mkdir /etc/polkit-1/rules.d/
cd /etc/polkit-1/rules.d/
wget https://pastebin.com/raw/144dRAbP
mv 144dRAbP 99-manjaro.rules
```

Auto-cpufreq config

```
cd /etc/
wget https://pastebin.com/raw/4ChzT2Tm
mv 4ChzT2Tm auto-cpufreq.conf
```

SYSTEM MAINTAINING

Remove broken symlinks

```
find . -xtype l -delete
```

Automate cache clearing

```
find ~/.cache/ -type f -atime +100 -delete
```

Remove all journalctl older than 2 weeks:

```
journalctl --vacuum-time=2weeks
```

Update, upgrade and clear cache

```
sudo pacman -Syu --noconfirm -q  
sudo pacman -Syyu --noconfirm -q  
paccache -rvk1  
yay -Syu --noconfirm  
yay --clean -Sc --noconfirm  
sudo snap refresh  
rm -rf /tmp/*  
reboot
```

TROUBLESHOOTING

If bluetooth is not working

```
sudo systemctl enable bluetooth && sudo systemctl start bluetooth
```

Have boot messages stay on screen

```
su
mkdir /etc/systemd/system/getty@tty1.service.d
nano /etc/systemd/system/getty@tty1.service.d/noclear.conf
[Service]
TTYVTDIsallocate=no
```

Unable to make AUR packages

Is /tmp full?

```
df -h /tmp
```

Clear it if it's full

```
rm -rf /tmp/*
```

Note: you can do a restart as well to clear this.

Failed services:

```
systemctl --failed
```

Logfiles:

```
journalctl -p 3 -b
```

Check packages that were installed but no packages depend on them now:

```
pacman -Qtd
```

Manually installed packages that no package depend on:

```
pacman -Qm
```

List installed packages:

```
pacman -Ql
yay -Ql
```

Touchpad not working:

```
yay -S xf86-input-libinput
```

Check connected usb devices:

```
yay -S xorg-xinput
xinput
```

Windows unmountable due to hibernation

Screen tearing

[Using compton](#)

[Using dmenu](#)

[Fixes specifically for NVIDIA](#)

Volume stuck on mute

Reactivating the screen backlight (BIOS ONLY)

```
sudo sed -i "s/\"(GRUB_CMDLINE_LINUX=\\)\\\"\"/\\1\"acpi_osi=Linux
acpi_backlight=vendor\\\"\" /etc/default/grub -i
sudo update-grub
```

Enable crontab

```
sudo pacman -S cronie
sudo systemctl enable cronie.service
sudo systemctl start cronie.service
```


USEFUL COMMANDS

Using SED to uncomment

Uncomment when text matches:

```
sudo sed -i '/*SEARCH_FOR_THIS*/s/^#/g' WHERE_TO_UNCOMMENT  
sudo sed -i '/autologin-user=hakszi/s/^#//g' /etc/lightdm/lightdm.conf
```

Uncomment at line X:

```
sed '*line_number*s/^#//' /etc/pacman.conf  
sed '93s/^#//' /etc/pacman.conf
```

Setup alias SSH connection with password

```
sshpass -p *PASSWORD_HERE* ssh *username*@*ip_address*  
sshpass -p hakszi ssh pi@pi.hole
```

Run the last command with sudo

```
sudo !!
```

UNSAFE COMMANDS

(without thinking)

- **pacman -Sy**
 - Note that this is essential when installing the system.
 - The bash script checkupdates, included with the pacman package, provides a safe way to check for upgrades to installed packages without running a system update at the same time.
 - TLDR: Refreshes package list, without upgrading the system, and this could lead to dependency issues.
- **rm -rf**
 - Doesn't ask, whether you want to uninstall the files, or not and deletes everything recursively. You can delete essential system files with this. Use safe-rm (yay -S safe-rm).
- **dd**
 - Informally known as disk destroyer (be very careful when using this)
- **chmod 777**
 - What does chmod 777 mean?
 - TLDR: Setting 777 permissions to a file or directory means that it will be readable, writable and executable by all users and may pose a huge security risk.

[Other dangerous commands](#)

GAMING

Getting steam set up:

```
sudo mkdir ~/.fonts/ && cd ~/.fonts/  
wget https://support.steampowered.com/downloads/1974-YFKL-4947/SteamFonts.zip  
unzip SteamFonts.zip && rm SteamFonts.zip
```

Mouse latency

```
sudo nano /etc/default/grub  
add:      usbhid.mousepoll=1  
to:       GRUB_CMDLINE_LINUX_DEFAULT  
update-grub
```

STEAM

[Steam configuration](#)

[Enable game mode scheduling \(performance improvement\)](#)

[Enable saving the Shader Cache](#)

[Laggy screen updates](#)

[using enviroment variables](#)

Set-up an AMD GPU for daily usage (Graphics,Gaming,Compute,etc) in Arch Linux.:

```
sudo nano /etc/environment  
AMD_VULKAN_ICD=RADV
```

Complied for certain games:

```
yay -S wine-ge-custom
```

Java (if another version is needed):

Find all available versions

```
sudo pacman -sS java | grep jdk
```

Install the one you want.

32 bit wine

[origin](#)

[origin on linux #2](#)

[Simcity](#)

Lutris install

SECURITY

[Block list for qbittorent](#) & original list can be found [here](#)

```
cd ~/.config/qBittorrent/  
download files from here into this folder.
```

Microcode

AMD

```
sudo pacman -Syy amd-ucode  
yay -S iucode-tool  
sudo update-grub
```

Intel

```
sudo pacman -Syy intel-ucode  
yay -S iucode-tool  
sudo update-grub
```

[Enforcing strong passwords](#)

[Running apps in sandbox \(firejail\)](#)

[Firejail \(manjaro\)](#)

[Running apps in sandbox \(polkit\)](#)

SAVING & RELOADING THE SYSTEM

Save:

```
fstrim /  
dd if=/partition/to/be/written/on bs=64M status=progress | zstd -q3T2c --adapt - > NAME.img.zst  
dd if=/dev/sda bs=64M status=progress | zstd -q3T2c --adapt - > arch.img.zst  
dd if=/dev/nvme0n1 bs=64M status=progress | zstd -q3T2c --adapt - > arch.img.zst
```

Reload previous save (from a live system):

```
zstd -vkdc /backup/file.img.zst | dd of=/dev/sdX bs=4M status=progress  
zstd -vkdc arch.img.zst | dd of=/dev/sda bs=4M status=progress  
zstd -vkdc arch.img.zst | dd of=/dev/nvme0n1 bs=4M status=progress
```

Speed up save and reload:

```
su  
cd /where/you/save/or/load/backup/from
```

Troubleshooting:

Problem #1: Terminal kills program, ram usage at 100%

Solution to #1: Extend ram, or (slower method):

```
sudo nano /etc/makepkg.conf  
MAKEFLAGS="-j2"
```

EXTRA PRODUCTIVENESS

Control 2 PC with one mouse and keyboard
Set up virtualbox on arch

Useful things to take a look at later (guessed right, procrastination)

https://wiki.archlinux.org/title/Frequently_asked_questions
https://wiki.archlinux.org/title/Improving_performance
https://wiki.archlinux.org/title/Improving_performance/Boot_process
https://wiki.archlinux.org/title/Security#Enforcing_strong_passwords_with_pam_pwquality

https://wiki.archlinux.org/title/Lenovo_V15_G2-ALC
https://wiki.archlinux.org/title/Lenovo_V15_G2-ALC#Function_Keys

https://wiki.archlinux.org/title/System_maintenance
<https://wiki.archlinux.org/title/Security>
https://wiki.archlinux.org/title/Domain_name_resolution#Privacy_and_security
https://wiki.archlinux.org/title/Power_management#Power_management_with_systemd
https://wiki.archlinux.org/title/CPU_frequency_scaling
https://wiki.archlinux.org/title/Solid_state_drive
https://wiki.archlinux.org/title/Uniform_look_for_Qt_and_GTK_applications

<https://wiki.archlinux.org/title/Backlight>
https://wiki.archlinux.org/title/Webcam_setup
https://wiki.archlinux.org/title/Hardware_video_acceleration#NVIDIA
<https://wiki.archlinux.org/title/Acpid>
https://wiki.archlinux.org/title/Extra_keyboard_keys
https://wiki.archlinux.org/title/Network_configuration/Wireless
https://wiki.archlinux.org/title/Display_Power_Management_Signaling
https://wiki.archlinux.org/title/CPU_frequency_scaling