ARCH LINUX "FROM SCRATCH" + GNOME + INTEL CHIPSET + UEFI + SECURE BOOT

1. UEFI & KEYBOARD & LOCALE

IMPORTANT: ENABLE SECURE BOOT WITH NO KEYS ENROLLED

a. ls /sys/firmware/efi/efivars ... if there's an error ... it's not UEFI
b. efibootmgr ... should list "Linux Boot Manager"

- loadkeys br-abnt2
- nano /etc/locale.gen ... unmark LANG=pt_BR.UTF-8
- locale-gen
- nano /etc/locale.conf ... set LANG=pt_BR.UTF-8

2. NETWORK & CONNECTIVITY BY WI-FI

- ip link
- iwctl
- device list
- station wlan0 scan
- station wlan0 get-networks
- station wlan0 connect
- exit
- ping 1.1.1.3
- timedatectl

- ls /usr/share/kbd/consolefonts/ | grep ter-122b
- if yes: setfont ter-122b
- if not: pacman -Sy terminus-font
- then : setfont ter-122b

3. PARTITIONING & FORMATTING & MOUNTING NVME

parted /dev/nvme0n1 mklabel gpt
parted /dev/nvme0n1 mkpart ESP fat32 1MiB 1025MiB
parted /dev/nvme0n1 set 1 esp on
parted /dev/nvme0n1 mkpart primary ext4 1025MiB 4097MiB
parted /dev/nvme0n1 mkpart primary btrfs 4097MiB 100%

mkfs.fat -F32 /dev/nvme0n1p1
mkfs.ext4 -L BOOT /dev/nvme0n1p2
mkfs.btrfs -L ROOT /dev/nvme0n1p3

mount /dev/nvme0n1p3 /mnt
btrfs subvolume create /mnt/@
btrfs subvolume create /mnt/@home
btrfs subvolume create /mnt/@log
btrfs subvolume create /mnt/@pkg
btrfs subvolume create /mnt/@snapshots

```
umount /mnt

mount -o subvol=@ /dev/nvme0n1p3 /mnt
mkdir -p /mnt/{home,var/log,var/cache/pacman/pkg,var/snapshots}
mount -o subvol=@home /dev/nvme0n1p3 /mnt/home
mount -o subvol=@log /dev/nvme0n1p3 /mnt/var/log
mount -o subvol=@pkg /dev/nvme0n1p3 /mnt/var/cache/pacman/pkg
mount -o subvol=@snapshots /dev/nvme0n1p3 /mnt/var/snapshots
mkdir -p /mnt/boot
mount /dev/nvme0n1p2 /mnt/boot
mkdir -p /mnt/boot/efi
mount /dev/nvme0n1p1 /mnt/boot/efi
```

4. BASE SYSTEM INSTALLATION FOR INTEL CHIPSET

```
reflector --latest 20 --sort rate --save /etc/pacman.d/mirrorlist

pacstrap -K /mnt base linux-firmware linux-lts linux-lts-headers linux-zen linux-zen-
headers dkms intel-ucode mkinitcpio vulkan-intel vulkan-mesa-device-select vulkan-
tools apparmor btrfs-progs efibootmgr iwd nano networkmanager sbctl sudo

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

5. CHROOT CONFIGURATION

```
arch-chroot /mnt
```

- ln -sf /usr/share/zoneinfo/America/Araguaina /etc/localtime
- hwclock --systohc
- nano /etc/locale.gen ... check if LANG=pt_BR.UTF-8 is unmarked
- locale-gen
- nano /etc/locale.conf ... set LANG=pt_BR.UTF-8
- nano /etc/vconsole.conf ... set KEYMAP=br-abnt2
- nano /etc/hostname ... set logos (choose a personal name)

- passwd ... setting "root" admin password
- useradd -m -g users -G wheel -s /bin/bash archer (choose a personal name)
- passwd archer ... setting archer sudo password
- EDITOR=nano visudo
- %wheel ALL=(ALL:ALL) ALL ... uncomment for enabling sudo for archer user

6. BOOTLOADER (SYSTEMD-BOOT)

a. bootctl config

```
blkid /dev/nvme0n1p3 ... catch the UUID in PARTUUID field

nano /boot/loader/entries/archer-zen.conf
```

```
title    Arch Linux ZEN
linux    /vmlinuz-linux-zen
initrd   /intel-ucode.img
initrd   /initramfs-linux-zen.img
options root=PARTUUID=<UUID catched> zswap.enabled=0 rootfstype=btrfs
rootflags=subvol=@ rw nvme_core.default_ps_max_latency_us=0 apparmor=1
security=apparmor loglevel=3


nano /boot/loader/entries/archer-lts.conf


title    Arch Linux LTS
linux    /vmlinuz-linux-lts
initrd   /intel-ucode.img
initrd   /initramfs-linux-lts.img
options root=PARTUUID=<UUID catched> zswap.enabled=0 rootfstype=btrfs
rootflags=subvol=@ rw nvme_core.default_ps_max_latency_us=0 apparmor=1
security=apparmor loglevel=3


IMPORTANT: mkinitcpio does not create or manage initramfs fallback


nano /boot/loader/loader.conf


default archer-zen.conf
timeout 2
console-mode max
editor no


b. sbctl config


- sbctl verify
- sbctl create-keys
- sbctl verify
- sbctl sign -s ... ( sign each entrie listed by verify command )
- sbctl verify
- sbctl enroll-keys
- sbctl list-enrolled-keys
- sbctl status


nano /etc/sysctl.d/99-sysctl.conf


kernel.kptr_restrict = 2
kernel.dmesg_restrict = 1
kernel.randomize_va_space = 2
fs.protected_hardlinks = 1
fs.protected_symlinks = 1


- sysctl --system


c. hooks config
```

mkdir -p /etc/pacman.d/hooks (must be!)

nano /etc/pacman.d/hooks/99-secureboot.hook

```
[Trigger]
Operation = Upgrade
Type = Package
Target = linux-zen
Target = linux-lts
Target = systemd

[Action]
Description = Signing linux-zen linux-lts kernels initramfs systemd-boot
When = PostTransaction
Exec = /bin/sh -c 'sbctl sign -s /boot/vmlinuz-linux-zen 2>/dev/null || true && sbctl
sign -s /boot/vmlinuz-linux-lts 2>/dev/null || true && sbctl sign -s /boot/initramfs-
linux-zen.img 2>/dev/null || true && sbctl sign -s /boot/initramfs-linux-lts.img
2>/dev/null || true && sbctl sign -s /boot/EFI/BOOT/BOOTX64.EFI 2>/dev/null || true
&& sbctl sign -s /boot/EFI/systemd/systemd-bootx64.efi 2>/dev/null || true'
```

d. generate initramfs

- ls /sys/firmware/efi/efivars ... check if UEFI remain active
- efibootmgr ... should list "Linux Boot Manager"

- systemctl enable apparmor.service
- systemctl enable NetworkManager.service

- mkinitcpio -P

- exit ... exiting from chroot environment

- umount -R /mnt
- swapoff -a

reboot + F2 + Secure Boot + Key Management + Verify Keys or Import Keys if you did
not has used sbctl enroll-keys + F10

7. POST-INSTALLATION NETWORK CONFIGURATION

Login: root

- systemctl start NetworkManager.service

- nmcli general status
- nmcli device status
- nmcli device wifi list
- nmcli device wifi connect "SSID" --ask (or -a)

- pacman -Sy terminus-font

- setfont ter-122b

- bootctl status

    Secure Boot: enabled (user keys)

- sbctl status

    Secure Boot: enabled
    Setup Mode: disabled
    Vendor keys: disabled
    User keys: enrolled

8. INSTALLING GNOME DESKTOP

[Gnome Desktop — 68] pacman -S adwaita-icon-theme colord eog extension-manager file-roller fwupd gdm gedit gedit-plugins gnome gnome-backgrounds gnome-bluetooth gnome-boxes gnome-browser-connector gnome-calculator gnome-calendar gnome-characters gnome-clocks gnome-color-manager gnome-control-center gnome-disk-utility gnome-firmware gnome-info-collect gnome-keyring gnome-logs gnome-menus gnome-music gnome-online-accounts gnome-power-manager gnome-session gnome-settings-daemon gnome-shell-extensions gnome-terminal gnome-text-editor gnome-themes-extra gnome-tweaks gnome-usage gnome-user-docs gnome-weather grilo-plugins gst-plugins-base gst-plugins-base-libs gthumb gvfs gvfs-afc gvfs-dnssd gvfs-goa gvfs-gphoto2 gvfs-mtp gvfs-nfs gvfs-smb gvfs-wsdd localsearch loupe mesa-utils mutter nautilus rygel seahorse simple-scan system-config-printer tinysparql xdg-desktop-portal-gnome xdg-user-dirs-gtk xkeyboard-config yelp yelp-tools yelp-xsl

- systemctl enable gdm.service
- systemctl status gdm.service

- reboot

Login: "user" & "password"

- sudo systemctl start gdm.service (typed if login was failed)

10. ESSENTIALS PACKAGES AND CONFIGURATIONS

Installation of applications and services complementary to Gnome Desktop

[GUI — 23] sudo pacman -S collision decibels firefox firefox-i18n-pt-br firewalld foliate font-manager fragments gparted eyedropper libreoffice-still libreoffice-still-pt-br mission-center mpv network-manager-applet papers pavucontrol ptyxis qalculate-gtk shotwell showtime snapshot zed

[CLI — 89] sudo pacman -S alsa-utils aspell aspell-en aspell-pt at-spi2-core avahi bashtop bat bind-tools bluez bluez-utils cronie cups cups-browsed cups-filters cups-pdf curl ethtool eza fail2ban fastfetch fd ffmpeg foomatic-db foomatic-db-engine foomatic-db-ppds fzf git glances grc gutenprint hspell htop hunspell hwinfo imagemagick inxi iproute2 less libssh libssh2 libvncserver libvoikko libwireplumber lsd lsof man-db man-pages meson ninja nmap nss-mdns ntfs-3g nuspell p7zip pacman pacman-contrib parted pipewire pipewire-pulse powertop pwgen qt5-wayland qt6-wayland

reflector ripgrep rpcbind smartmontools speech-dispatcher speedtest-cli tecla tlp tree unzip upower v4l-utils v4l2loopback-utils wget wireless_tools wireplumber zip zram-generator zsh zsh-autocomplete zsh-autosuggestions zsh-completions zsh-history-substring-search zsh-lovers zsh-syntax-highlighting

[Sistem Fonts – 48] sudo pacman -S gnu-free-fonts terminus-font ttf-anonymous-pro ttf-atkinson-hyperlegible ttf-bitstream-vera ttf-caladea ttf-carlito ttf-cascadia-code ttf-crimson-pro ttf-crimson-pro-variable ttf-croscore ttf-dejavu ttf-doulos-sil ttf-droid ttf-eurof ttf-fantasque-sans-mono ttf-fira-code ttf-fira-mono ttf-fira-sans ttf-hack ttf-ibm-plex ttf-inconsolata ttf-input ttf-jetbrains-mono ttf-jetbrains-mono-nerd ttf-junicode ttf-junicode-variable ttf-khmer ttf-lato ttf-liberation ttf-libertinus ttf-linux-libertine ttf-linux-libertine-g ttf-material-icons ttf-material-symbols-variable ttf-meslo-nerd ttf-mona-sans ttf-monaspace-frozen ttf-monaspace-variable ttf-monofur ttf-monoid ttf-montserrat ttf-nunito ttf-opensans ttf-overpass ttf-roboto ttf-roboto-mono ttf-terminus-nerd

- sudo systemctl enable avahi-daemon.service
- sudo systemctl start avahi-daemon.service
- sudo systemctl enable bluetooth.service
- sudo systemctl start bluetooth.service
- sudo systemctl enable cronie.service
- sudo systemctl start cronie.service
- sudo systemctl enable cups.service
- sudo systemctl start cups.service
- sudo systemctl enable cups-browsed.service
- sudo systemctl start cups-browsed.service
- sudo systemctl enable fstrim.timer
- sudo systemctl start fstrim.timer
- sudo systemctl enable tlp.service
- sudo systemctl start tlp.service

1. setup gnome-control-center
2. setup gnome-tweaker
3. setup extension-manager
4. setup fstab: sudo nano /etc/fstab
5. setup zram-generator
6. setup zsh: chsh -s /bin/zsh $USER ... then logout & login

yt-dlp:

sudo wget -O /usr/local/bin/yt-dlp https://github.com/yt-dlp/yt-dlp/releases/latest/download/yt-dlp && sudo chmod a+rx /usr/local/bin/yt-dlp