# ARCH LINUX PARA INICIANTES

# COMO INSTALAR O SISTEMA PASSO A PASSO!

* USB BOOTABLE
* Para criar um USB bootable usando o comando (dd) no Linux:

**dd bs=4M if='/lugar\_onde\_esta\_seu\_iso' of='/lugar\_para\_o\_qual\_copiar' status=progress oflag=sync**

(Substitua o **X** pela letra do seu dispositivo ex: 'sdc' 'sdd') use: **fdisk -l**

* Windows/Linux use o **Etche**r ou **Rufus**
* DOCUMENTAÇÃO

Você pode consulta a documentação no **live cd**, basta alternar entre **ALT+F1*(tty1)*** e **ALT+F2*(tty2)***, seguindo os passos necessários para instalação

É necessário loga como Root: **root > ls > nano install.txt**

* **TECLADOR, LOCALIZAÇÃO, IDIOMA E RELÓGIO:**

1. Definir o layout do teclado para abnt2 (PT-BR) localectl list-keymaps | grep BR

Lista layouts disponíveis: **ls /usr/share/kbd/keymaps/\*\*/\*.map.gz**

loadkeys br-abnt2 **ou** localectl set-keymap br-abnt2

1. Aumentar a fonte do console(Fontes disponíveis: **ls /usr/share/kbd/consolefonts/)**

**setfont lat4-19 setfont lat0–16**

ou **pacman -Syy terminus-font && setfont ter-132n**

1. Vamos alterar o idioma da instalação:

Edite o arquivo de geolocalização, **nano /etc/locale.gen** , aperte “**Ctrl + W**” para abrir um campo de pesquisa, digite “**pt\_BR**” e dê Enter para iniciar pesquisa.

Configure o locale (**localização**), descomentando essa linha: **pt\_BR.UTF-8 UTF-8 e** salve “**Ctrl+o**” depois (**Enter**) e sair “**Ctrl+x**”

1. **Após o procedimento de alteração:**

**locale-gen && export LANG=pt\_BR.UTF-8**

1. Atualize o relógio do sistema

**timedatectl set-ntp true**

**timedatectl status** (verificar o status do service)

\*\*\* Estas configurações só são **válidas somente para o Modo Live** é necessário reconfigurar quando fizer o **CHROOT**

### VERIFIQUE O MODO DE INICIALIZAÇÃO EM BIOS OU UEFI

Se este comando a seguir listar as variáveis **EFI**, isso significa que você iniciou a operação com sucesso no **modo EFI**. Caso o diretório não exista, o sistema pode ser inicializado no modo Legacy BIOS, **legacy-mode**.

**ls /sys/firmware/efi/efivars**

Se o diretório não existir, o sistema pode ser inicializado no modo **BIOS** ou **CSM**.

### CONEXÃO COM A INTERNET

1. Interface de rede:

**ip link** ou **ip addr ou ip a**

1. Internet cabeada, por padrão, temos o DHCP ativado no sistema, se necessário for inicie sua placa de rede.

**systemctl start dhcpcd**

1. Caso sua rede seja wireless.

* Teste a conectividade:

**ping -c 3** [www.google.com](http://www.google.com/)

**iwctl**

1. **device list (**liste todos os dispositivos Wi-Fi)
2. **station wlan0 scan (**procurar redes)
3. **station wlan0 get-networks (**listar todas as redes disponíveis )
4. **station wlan0 connect “SSID” (**conectar-se a uma rede)
5. **iwctl --passphrase “SENHA” station wlan0 connect “SSID”**

* PARTICIONAMENTO DE DISCO

**Decida o tipo de tabela de partição**

* BIOS ou UEFI(você precisa usar GPT)

**Recomendações de tamanho de Sistema EFI**

* Pelo menos: 150 MB
* Recomendado: 300 MB

**GPT (UEFI)**

|  |  |  |  |
| --- | --- | --- | --- |
| Precisava | Partição | Tipo de partição | Ponto de montagem |
|  | /dev/sdXY | Partição do sistema EFI | /mnt/boot/efi |
|  | /dev/sdXY | Troca de Linux | - |
|  | /dev/sdXY | Linux | /mnt |
|  | /dev/sdXY | Linux | /mnt/home |

**DOS (BIOS)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Precisava | Partição | Tipo de partição | Ponto de montagem | Bandeiras |
|  | /dev/sdXY | Troca de Linux | - | - |
|  | /dev/sdXY | Linux | /mnt | Inicializável |
|  | /dev/sdXY | Linux | /mnt/home | - |

**SWAP**

|  |  |  |
| --- | --- | --- |
| RAM do sistema | Espaço de troca recomendado | Se estiver usando hibernação |
| <2 GB | 2x a quantidade de RAM | 3x a quantidade de RAM |
| 2-8 GB | Igual à quantidade de RAM | 2x a quantidade de RAM |
| 8-64 GB | Pelo menos 4 GB | 1,5x a quantidade de RAM |
| 64 GB | Pelo menos 4 GB | Hibernação não recomendada |

1. Descobri qual disco rígido foi detectado:  
   **dmesg | grep sd ou lsblk**

Você é livre para definir o particionamento da maneira que for mais conveniente, e terá à sua disposição ferramentas como **fdisk***(Baseado em texto)*, **gdisk***(UEFI apenas com base em texto)*, **cfdisk***(Gráfico, recomendado para iniciantes)*, **cgdisk***(UEFI apenas gráfico*, *recomendado para iniciantes)* e **parted**, que podem ser acessadas digitando esses nomes.

* **Layout**

**/dev/sda1 (300MB para o /boot/efi)**

**/dev/sda2 (45GB para /)**

**/dev/sda3 (4GB para swap)**

**/dev/sda4 (todo o resto para o /home)**

**Escrever/Write as mudanças nas partições.**

**Escrever/Write — Sim/Yes**

**Quit/Sair**

1. Vamos usar **cfdisk** para fazer o particionamento o disco**, é bem mais intuitivo.**

**cfdisk /dev/sda**

Vamos ter que criar 3 ou 4 partições (4 no caso e você querer usar uma partição como **SWAP** ao invés de um **SWAPFILE**). Caso seja necessário, você deve criar uma nova tabela a partição, escolha **GPT** se você usar **UEFI**, se for usar **BIOS-legacy**, pode ser **MBR**.

Lembre-se de fazer o **cfdisk** **escrever** as partições e de marcar a partição que receberá o **GRUB** (no meu caso a **/dev/sda1**/ como partição **UEFI** em “**type**”).

### FORMATAR AS PARTIÇÕES

1. Na sequencia formate o **EFI**, **Raiz**, **Swap** e **Home**

**mkfs.fat -F32 –n EFI /dev/sda1** (Caso não exista uma paratição ESP) **&& mkfs.ext4 –L RAIZ /dev/sda2 && mkswap -L SWAP /dev/sda4 && mkfs.ext4 –L HOME /dev/sda3**

1. **Verificando as UUID(s) e as Label(s) das partições recém-formatadas: blkid ou lsblk -f**

### CRIAR PONTOS DE MONTAGEM E MONTANDO AS PARTIÇÕES

O próximo passo é fazer a montagem das partições do sistema, atente-se que será necessário criar algumas pastas para poder fazer a montagem.

1. Primeiro, monte a partição **raíz** em **/mnt** e depois ative a **swap**:

**mount /dev/sda2 /mnt *&&* swapon /dev/sda4**

Para verificar se a partição **swap** está funcionando: **free -h** ou **swapon -s**

1. Crie e monte o diretório **/home**:

* Criando todos os diretórios com único comando, depois é só montagem:

**mkdir –p /mnt/{home,boot/efi}**

**mkdir /mnt/home *&&* mount /dev/sda3 /mnt/home**

1. No caso de uma instalação em **DualBoot** com o **Windows,** so necessita montar a partição ***EFI System)***:

**mkdir -p/mnt/boot/efi *&&* mount /dev/sdaX /mnt/boot/efi** (se for usar UEFI***)***

1. Verificar o layout de particionamento atual: **lsblk /dev/sdX ou lsblk -fm ou df -h**

* ESCOLHER/CLASSIFICAR O ESPELHO DE DOWNLOAD

Alterar os mirros do Arch para quem sabe, fazer o download mais rápido, esse passo é opcional, mas pode ser feito editando o arquivo (estou usando o editor ‘nano’ nesse caso):

**nano /etc/pacman.d/mirrorlist**

Você pode mover o mirror desejado para cima, ou simplesmente comentar com um **#** os que você não quiser que sejam usados.

Pressione “**Ctrl + W**” para pesquisarmos os espelhos. Digite **BRAZIL** e dê um **Enter**.

Recorte e cole no topo da lista, as linhas com os endereços dos servidores brasileiros com o “**Ctrl+K**” e “**Ctrl+U**” respectivamente.

Dê um “**Ctrl+O**” para salvar e “**Ctrl+X**” para fechar o arquivo. Depois **pacman -Syyy**

Se desejar classifica e ordenar de forma detalhada os 10 servidores mais rápidos, com suporte a HTTPS e HTTP e que foram atualizados nas últimas 24 horas e **sobrescreve** o arquivo **/etc/pacman.d/mirrorlist**:

**sudo pacman -Sy reflector && sudo** **reflector -l 10 -a 24 -p https,http --sort rate --save /etc/pacman.d/mirrorlist**

**reflector -c Brazil -l 5 -a 24 --sort rate --save /etc/pacman.d/mirrorlist && pacman -Syy**

**Ou**

1. Instalação & Backup:

**pacman -S pacman-contrib && cp /etc/pacman.d/mirrorlist /etc/pacman.d/mirrorlist.backup**

1. Descomentar todos os espelhos

**sed -i 's/^#Server/Server/' /etc/pacman.d/mirrorlist.backup**

1. Classificando apenas os 5 espelhos mais rápidos: ( Isso pode demorar um pouco)

**rankmirrors -n 5 /etc/pacman.d/mirrorlist.backup > /etc/pacman.d/mirrorlist**

### INSTALAR OS PACOTES BASE DO ARCH LINUX

O próximo passo é a instalação dos pacotes **base** do Arch nas partições que criamos para instalar o sistema no diretório **sda2**(RAIZ), como também alguns programas necessários.

* **Escolha do Kernel:**

**pacman -S linux** (defaut, stable)

**pacman -S linux-lts** (long term support)

**pacman -S linux-hardened** (security features)

**pacman -S linux-zen** (base Zen Debian)

**pacstrap /mnt base base-devel linux linux-firmware** **linux-headers sudo nano git reflector xdg-utils xdg-user-dirs bash-completion man-db man-pages-pt\_br** **terminus-font sshfs openssh tmate**

Para conferir se tudo está ok? Verifique com um **ls** se as pastas foram criadas em /mnt.

### CONFIGURAR O FSTAB

Feita essa parte, vamos gerar a nossa tabela FSTAB, que vai dizer para o sistema onde estão montadas cada uma das partições, faremos isso usando este comando:

**genfstab -U /mnt >> /mnt/etc/fstab** (Gerando)

* Esse “**-U**” ali “no meio da turma” é para que sejam usados os **IDs** dos discos no **FSTAB**, ao invés dos rótulos.

FSTAB: **Dev, Partuuid(-U) e Label(-L)™**

Consulta as UUID(s) das partições: **blkid**

Você pode verificar se o arquivo foi gerado com sucesso e o seu conteúdo com:

**cat /mnt/etc/fstab**

### NOVO SISTEMA (chroot, alterar raiz)

[Mude a raiz](https://wiki.archlinux.org/index.php/Change_root_(Português)) para novo sistema usando o ambiente **BASH**:

**arch-chroot /mnt**

Uma vez logado no seu sistema (**repare que o terminal mudou de aparência**), tudo o Tudo que você fizer agora, ficará em definitivo no seu Arch Linux.

* Você pode alterar data e hora depois, quando instalarmos uma interface, assim como o fuso horário, mas se você quiser fazer isso agora, manualmente (como manda a ‘bíblia’)

1. Configurações de Idioma e gerar o código da localidade

**echo pt\_BR.UTF-8 UTF-8 >>** **/etc/locale.gen && locale-gen**

1. Configurar a variável de linguagem em locale.conf, usando este comando:

**echo “LANG=****pt\_BR.UTF-8” > /etc/locale.conf**

1. Configurar Keymap(Mapa de Teclado), layout

**echo -e "KEYMAP=br-abnt2” >** **/etc/vconsole.conf**

1. Configurar teclado abnt2 no ambiente X: **localectl set-x11-keymap br abnt2** ou **setxkbmap -model abnt2 -layout br -variant abnt2**
2. Consutar os fusos horários disponíveis da América: **ls -a /usr/share/zoneinfo/America**

**ou timedatectl list-timezones | grep Recife**

**timedatectl set-time-zone America/Recife**

**ln -sf /usr/share/zoneinfo/America/Recife /etc/localtime --force**

1. Sincronização do Relógio com Hardware (BIOS), gerar /etc/adjtime

**hwclock --systohc ~~--utc~~** (UTC padrão) \*\*\*Conferindo data, hora e fuso horário: **date**

* DEFINIR HOSTNAME, CONFIGURAR O ARQUIVO HOSTS e DNS

1. Dar um nome para a máquina

**echo “hostname” > /etc/hostname**

Ou **nano /etc/hostname**

Ou **hostnamectl set-hostname meuhostname**

1. Com o hostname definido, abra o arquivo /etc/hosts:

**nano /etc/hosts**

Adicione entradas correspondentes aos hosts:

Em “**meuhostname**”, nesse caso, seria “**NomedaMaquinaEscolhido**”, sem aspas.

|  |  |
| --- | --- |
| * Alterar DNS: | |
| **nano /etc/resolv.conf** | |
| nameserver | 8.8.8.8 |
| nameserver | 8.8.4.4 |

|  |  |  |
| --- | --- | --- |
| **127.0.0.1** | **localhost.localdomain** | **localhost** |
| **::1** | **localhost.localdomain** | **localhost** |
| **127.0.1.1** | **meuhostname.localdomain** | **meuhostname** |

### Se o sistema tem um endereço IP permanente, ele deve ser usado em vez de **127.0.1.1**.

### CONFIGURANDO A CONEXÃO

1. Ethernet:

**pacman - S networkmanager network-manager-applet**

**systemctl enable dhcpcd (Escolha entre habiltar dhcpcd ou NetworkManager)**

**systemctl enable NetworkManager && systemctl start NetworkManager**

**systemctl status NetworkManager**

1. Wifi:

**pacman -S wireless\_tools wpa\_supplicant dialog ifplugd net-tools netctl networkmanager network-manager-applet**

**systemctl enable NetworkManager && systemctl start NetworkManager**

**systemctl status NetworkManager**

* ~~INITRAMFS~~

~~Um script bash usado para criar um ambiente~~[~~disco RAMDISK inicial~~](https://en.wikipedia.org/wiki/Initial_ramdisk) **~~: mkinitcpio -P~~**

* SENHA DO ROOT **passwd**
* INSTALAR BOOT-LOADER (GRUB)
* Para que outros sistemas operacionais sejam reconhecidos e apareça na lista do Grub, instale o pacote “**os-prober**” antes de gerar o **grub.cfg** e se você tiver uma **CPU Intel**, instale o pacote **intel-ucode,** vale salietar a importancia de instalar tambem alguns utilitários para gerenciar o sistema de arquivos e um agente de autenticao

**pacman -S os-prober intel-ucode dosfstools mtools** **freetype2 libisoburn fuse2 ntfs-3g e2fsprogs gvfs-mtp gvfs-gphoto2 gvfs-afc gvfs-smb udisks2 polkit-gnome ifuse**

|  |  |
| --- | --- |
|  | [polkit-gnome](https://archlinux.org/packages/community/x86_64/polkit-gnome/) |
|  | [gvfs-afc](https://archlinux.org/packages/extra/x86_64/gvfs-afc/) |

### **MODO BIOS**

|  |
| --- |
| 1. Instale o pacote necessário: |
| **pacman -S grub** |
| 1. Instalando o Grub2 na partição de boot: |
| **grub-install --target=i386-pc --recheck /dev/sdX** |
| 1. ~~Para evitar uma mensagem de erro (inofensiva) no momento da inicialização:~~ |
| **~~mkdir /boot/grub/locale~~** |
| **~~cp /usr/share/locale/en\@quot/LC\_MESSAGES/grub.mo/boot/grub/locale/en.mo~~** |
| 1. Gerando o grub.cfg: |
| **grub-mkconfig -o /boot/grub/grub.cfg** |

* **MODO UEFI**

|  |
| --- |
| 1. Instale os pacotes necessários: |
| **pacman -S grub efibootmgr** |
| Instalando o grub e checando os S.O iniciáveis, substitua $esp por /boot/efi ou equivalente: |
| **grub-install --target=x86\_64-efi --efi-directory=/boot/efi --bootloader-id=grub --recheck** |
| 1. ~~Para evitar uma mensagem de erro (inofensiva) no momento da inicialização:~~ |
| **~~mkdir /boot/grub/locale~~** |
| **~~cp /usr/share/locale/en\@quot/LC\_MESSAGES/grub.mo /boot/grub/locale/en.mo~~** |
| 1. Gerando o arquivo de configuração do grub:   **grub-mkconfig -o /boot/grub/grub.cfg** |

### DESMONTAR AS PARTIÇÕES E REINICIAR

### Saia do ambiente chroot, desmontando as partições e reiniciando para poder ir para o próximo passo, a pós instalação.

### **Exit ou CTRL+D**

### **umount -R /mnt** ou **umount -a**

**reboot** ou **shutdown -r now**

PÓS INSTALAÇÃO

* ACESSO REMOTO(SSH, Instalação por outro PC, facilitando o copiar e colar de comandos)

**systemctl enable sshd && systemctl start sshd** (Ativando e iniciando o serviço no PC da instalação)

**ssh user@ip –p xx** (acesso em outro PC para dá continuidade a instalação**)**

* MICROCODE (Foi atualizado no boot?)

**dmesg | grep microcode**

* CRIAR USUÁRIO

Você pode criar outros usuários se você quiser agora, mas isso também pode ser feito depois, pela interface, o que acaba sendo mais simples. De toda forma, como exemplo, você pode criar um usuário com o seu nome.

1. Criando usuário:

* **OPÇÕES DO USERADD**

**useradd -m -G [initial\_group] -G [additional\_groups] -s [login\_shell] [username]**

**useradd -mG wheel,storage,network,autologin,vboxusers,libvirt,adbusers,plugdev *username***

**-m: Cria o diretório home fornecido pela opção "-d" com os arquivos e estrutura definidas na configuração de /etc/skell**

**-G *grupos*: especifica grupos suplementares, separados por uma vírgula, sem espaços em branco. O usuário pode fazer parte de mais de um grupo sem ser removido de outros.**

**Listar grupos do usuário: groups $USER**

**Criar um novo grupo**: **sudo groupadd** “grupo”

**Adicionar usuário ao grupo sudo** **gpasswd -a** **$USER** “grupo” OU **sudo usermod -aG** “grupo” **$USER**

**-a Adicione o usuário ao(s) grupo(s) suplementar(es) sem remover de outros grupos. Use apenas com a opção -G .**

1. Em seguida, forneça a senha para este novo usuário executando:

**passwd username**

Se você deseja forçar seu usuário a alterar a senha após o primeiro login:

**chage -d 0 yourusername**

1. Depois basta rodar o seguinte comando para criar as pastas de usuário:

**xdg-user-dirs-update**

Permitir que os usuários no grupo wheel, sejam capazes de executar tarefas administrativas com o sudo: **sed -i '/%wheel ALL=(ALL) ALL/s/^#//' /etc/sudoers**

Ou pelo **nano /etc/sudoers ,** descomentando(**#**) a declaração do grupo **wheel** como mostrado a seguir:

# Uncomment to allow members of group wheel to execute any command

**%wheel ALL=(ALL) ALL**

Ou **EDITOR=nano visudo**

1. Dados de usuário:

**usermod –c ‘Nome Completo’ $USER**

* REDE

Conexão **Wired** é reconhecida automaticamente. já **Wireless:**

1. Logue com seu username e senha (SuperUsuário #)

**su**

1. Conecte a sua rede **Wi-Fi** (Caso tenha, você pode usar **nmtui** ou **wifi-menu**)

**nmtui**

1. Verificar a conectividade com a net:

**ping -c 3 www.google.com**

* INSTALAR DISPLAY SERVER (Sistema de janelas X)

Um **display server ou servidor de janela** é um programa cuja principal tarefa é coordenar a entrada e saída de seus clientes para o sistema operacional, o hardware e entre eles. Em outras palavras, o **display server** controla e gerencia os recursos de baixo nível para ajudar a integrar as partes da **GUI**. Por exemplo, os **display server** gerenciam o mouse e ajudam a combinar os movimentos do mouse com o cursor e os eventos **GUI** causados pelo cursor. Mas não se confunda, o servidor de exibição não desenha nada. Eles apenas gerenciam a interface, as bibliotecas, os toolkits e, como você pode ver, eles se comunicam diretamente com o **kernel**.

**pacman -S xorg xorg-xinit** **mesa**

### INSTALAR DRIVERS GRÁFICOS

É hora de instalar drivers de vídeo. Eu suponho que você sabe qual GPU você está usando. Se você não sabe qual drive de vídeo você possui, descubra com esse comando:

**lspci -k | grep -A 2 -E "(VGA|3D)"**

Instale o que for referente ao seu **pacman -Ss** **xf86-video- ?** :

f

**pacman -S** **virtualbox-guest-utils virtualbox-host-modules-arch** (para VirtualBox)

**pacman -S xf86-video-fbdev** (para Hyper-V)

**pacman -S xf86-video-vmware** (VMWare)

**pacman -S xf86-video-qxl** (para Qemu)

**pacman -S xf86-video-intel** (para placas da Intel)

**pacman -S xf86-video-amdgpu** (GPUs AMD mais recentes)

**pacman -S xf86-video-ati** (GPUs AMD mais antigas)

**pacman -S nvidia nvidia-utils** (para placas Nvidia) #Proprietário

**pacman -S xf86-video-nouveau** (para placas Nvidia) #OpenSourceDriver

* INSTALAR DESKTOP ENVIRONMENT OU WINDOW MANAGER

Depois de instalar o **SERVIDOR X** você precisa de um **AMBIENTE** seja ele um **Desktop Environment(Ambiente de Trabalho)** ou um **Window Manager**(**Gerenciador de Janelas)** para fazer seus trabalhos diários!

* **DESKTOP ENVIRONMENT(AMBIENTE DE TRABALHO):**

**Xfce4** Desktop Environment: **# pacman -S xfce4 xfce4-goodies**

**Budgie** Desktop Environment: **# pacman -S budgie-desktop**

**GNOME** Desktop Environment: **#** **pacman -S gnome gnome-extra**

**Cinnamon** Desktop Environment: **#** **pacman -S cinnamon nemo-fileroller**

**KDE** Desktop Environment: **# pacman -S plasma-desktop kdebase**

**Mate** Desktop Environment: **# pacman -S mate mate-extra**

**Deepin** Desktop Environment: **# pacman -S deepin deepin-extra**

**Enlightenment** Desktop Environment: **# pacman -S enlightenment**

**LXDE** Desktop Environment: # **pacman -S lxde lxdm-gtk3**

**LXQt** Desktop Environment: # **pacman -S lxqt**

Eles fornecem uma interface de usuário gráfica completa para um sistema, agrupando uma variedade de clientes do servidor de exibição.

* **WINDOW MANAGER**(**GERENCIADOR DE JANELAS):**

Os gerenciadores de janela são **CLIENTES X** que controlam a aparência e o comportamento dos quadros ("janelas"), onde os vários aplicativos gráficos são desenhados. Eles determinam a borda, a barra do título, o tamanho e a capacidade de redimensionar as janelas, e muitas vezes fornecem outras funcionalidades, como áreas reservadas para aderir dockapps como o Window Maker, ou a capacidade para separar janelas como o Fluxbox. Alguns gerenciadores de janela são empacotados com utilitários simples, como menus para iniciar programas ou para configurar o próprio **WM**.

I3WM: **pacman -S i3**

BSPWM**: pacman -S bspwm sxhkd**

DWM: **pacman -S dwm**

AWESOME: **pacman -S awesome**

XMONAD: **pacman -S xmonad**

QTILE: **pacman -S qtile**

HERBSTLUFTWM**: pacman -S herbstluftwm**

* DISPLAY MANAGER OU LOGIN MANAGER

Se você estiver instalando o **Desktop Environment(D.E)** de sua escolha, você notará que não existe um ambiente de **login gráfico**(**Login Manager)**. Então, isso significa que você pode fazer **login usando a linha de comando** e, em seguida, iniciar o **D.E** ou instalar um **gerenciador de login**, que após um **login** bem-sucedido, iniciará automaticamente o **D.E** para você.

Vale salientar a importância do arquivo **XINIT,** tendo em vista que o servidor de exibição **XORG** vai ser iniciado automaticamente.

* Instalar, Iniciar & Habilitar LightDM Display Manager

**sudo pacman -S lightdm lightdm-gtk-greeter lightdm-gtk-greeter-settings**

**systemctl start lightdm && systemctl enable lightdm**

* Instalar, Iniciar & Ativar lxdm Display Manager

**sudo pacman -S lxdm**

**sudo systemctl start lxdm && sudo systemctl enable lxdm**

* Instalar, Iniciar & Ativar GDM Display Manager

**sudo pacman -S gdm**

**sudo systemctl start gdm** **&&** **sudo systemctl enable gdm**

* Instalar, Iniciar & Ativar SDDM Display Manager

**sudo pacman -S sddm sddm-kcm**

**sudo systemctl start sddm && sudo systemctl enable sddm**

* LOGIN MANAGER + CONFIG FILE XINIT
* XFCE

**pacman -S xfce4 lightdm lightdm-gtk-greeter**

**echo "exec startxfce4" > ~/.xinitrc**

**systemctl enable lightdm**

* PLASMA 5

**echo "exec startkde" > ~/.xinitrc**

**sudo pacman -S plasma-desktop**

**sudo pacman -S sddm && systemctl enable sddm**

* GNOME

**echo "exec gnome-session" > ~/.xinitrc**

**sudo pacman -S gnome**

* **Cinnamon**

**echo "exec cinnamon-session" > ~/.xinitrc**

**sudo pacman -S cinnamon mdm**

**systemctl enable mdm**

* **Mate**

**echo "exec mate-session" > ~/.xinitrc**

**sudo pacman -S mate lightdm lightdm-gtk-greeter**

**systemctl enable lightdm**

* **Unity**

**A instalação do Unity é complicada - consulte o**[**Arch Linux Wiki**](https://wiki.archlinux.org/index.php/Main_page_(Português))**.**

* Budgie

**echo "export XDG\_CURRENT\_DESKTOP=Budgie:GNOME" > ~/.xinitrc**

**echo "exec budgie-desktop" >> ~/.xinitrc**

**sudo pacman -S budgie-desktop lightdm lightdm-gtk-greeter**

**systemctl enable lightdm**

* Openbox

**echo "exec openbox-session" > ~/.xinitrc**

**sudo pacman -S openbox lightdm lightdm-gtk-greeter**

**systemctl enable lightdm**

* Deepin

**echo "exec startdde" > ~/.xinitrc**

**sudo pacman -S deepin deepin-extra**

Além disso, edite o arquivo /etc/lightdm/lightdm.conf para ter esta linha:

**greeter-session=lightdm-deepin-greeter**

* LXDE

**echo "exec startlxde" > ~/.xinitrc**

**sudo pacman -S lxdm-gtk3 lxdm**

* i3

**echo "exec i3" > ~/.xinitrc**

**sudo pacman -S i3-wm**

* Awesome

**echo "exec awesome" > ~/.xinitrc**

**sudo pacman -S awesome**

* TESTAR SERVIDOR X(Iniciar GUI)

Agora para ver se tudo está dando certo digita: **startx ou startx /usr/bin/(Desktop Environment Ou Window Manager)**

Agora pressione **CTRL+D** para sair

* ADVANCED LINUX SOUND ARCHITECTURE (ALSA)

**pacman -S alsa-utils alsa-lib alsa-firmware alsa-plugins pulseaudio pulseaudio-alsa pavucontrol sof-firmware(Dell Latitude 3410)**

Se não funcionar, reinicie o pulseaudio : **sudo systemct --user restart pulseaudio.service**

* BLUETOOTH

1. Bluetooth é necessário instalar em alguns pacotes e iniciar o serviço de Bluetooth:

**pacman -S bluez blueman bluez-utils pulseaudio-bluetooth && modprobe btusb**

1. Habilitar e iniciar o serviço de bluetooth:

**systemctl enable bluetooth && sudo systemctl start bluetooth**

* SERVIÇO DE IMPRESSÃO

**pacman -S ghostscript cups gsfonts libcups print-manager system-config-printer simple-scan gutenprint foomatic-db-engine epson-inkjet-printer-escpr epson-inkjet-printer-escpr2** ~~skanlite hplip~~

**systemctl enable cups.service** **&&** **systemctl start cups.service**

**systemctl list-unit-files | grep cups (status de serviços do cups)**

* LAPTOP?

1. Instale o **ACPI** gerenciamento inteligente de energia em seu sistema:

**pacman -S acpi acpid**

1. Drivers do seu touchpad, mouse e teclado, respectivamente:

**pacman -S xf86-input-libinput xf86-input-synaptics xf86-input-evdev ~~xf86-input-mouse xf86-input-keyboard~~**

* FONTES

**yay -S ttf-dejavu ttf-liberation ttf-font-awesome ttf-font-icons ttf-ionicons siji-git ttf-fantasque-sans-mono adobe-source-sans-pro-fonts ttf-opensans noto-fonts freetype2 terminus-font ttf-bitstream-vera ttf-droid ttf-fira-mono ttf-fira-sans ttf-freefont ttf-inconsolata libertinus-font**

Se você quiser, pode instalar todas as fontes disponíveis, o que ocupa cerca de 1 GB de espaço: **yay -S all-repository-fonts**

* USANDO FONTES DO WINDOWS

**sudo mkdir /usr/share/fonts/WindowsFonts**

**sudo cp /windows/Windows/Fonts/\* /usr/share/fonts/WindowsFonts/**

**sudo chmod 644 /usr/share/fonts/WindowsFonts/\***

**fc-cache -f**

Ou **yay –S ttf-ms-win10**

Ou **cd /tmp** **&&** **git clone** [**https://aur.archlinux.org/ttf-ms-win10.git && cd ttf-ms-win10**](https://aur.archlinux.org/ttf-ms-win10.git%20&&%20cd%20ttf-ms-win10) **&&** **makepkg -rsi --skipchecksums**

Leia **PKGBUILD** e copie todos os arquivos do Windows para o diretório e execute:

* MACOS FONTS

Se você quiser a fonte San Francisco da Apple

**yay -S --mflags --skipchecksums otf-san-francisco otf-san-francisco-mono**

* DICIONÁRIOS/ VERIFICADORES ORTOGRÁFICOS **yay –S** **hunspell-pt-br**
* WALLPAPER **(/usr/share/backgrounds)** **sudo pacman -S archlinux-wallpaper**

## NANO SINTAXE DESTACADA

Este pacote fornece aprimoramentos de realce de sintaxe para o editor de texto nano

**yay -S nano-syntax-highlighting**

## TEMA QT NO GTK DESKTOP

Se você usa uma área de trabalho GTK e deseja que os aplicativos Qt usem seu tema GTK, você tem 2 opções:

Você precisa instalar os plug-ins de estilo Qt5 para ambos

**yay -S qt5ct qt5-styleplugins**

### ~~Converter tema GTK2~~

~~Isso pode não parecer bom com todos os temas GTK~~

**~~echo "export QT\_QPA\_PLATFORMTHEME=gtk2" >> ~/.profile~~**

### Converte tema Qt5

**echo "export QT\_QPA\_PLATFORMTHEME=qt5ct" >> ~/.profile**

Ou

**echo "export QT\_QPA\_PLATFORMTHEME=qt5ct"** **>> /etc/environment**

**echo “****export QT\_AUTO\_SCREEN\_SCALE\_FACTOR=0” >> /etc/environment**

* COMPACTADORES E DESCOMPACTADORES

**yay -S file-roller cabextract zip lhasa rar p7zip unrar unarchiver unzip unace xz arj unarj tar gzip bzip2 ncompress** ~~xarchiver engrampa ark squeeze~~

* CODECS DE MIDIA

**yay -S gst-libav gst-plugins-base gst-plugins-good gst-plugins-bad gst-plugins-ugly gstreamer-vaapi gst-transcoder x265 x264 lame**

## PARTIÇÕES APFS

Se você tiver uma instalação do Hackintosh, pode usá-la para acessar seus arquivos a partir dele

**yay -S linux-apfs-dkms-git**

* SSD

**sudo systemctl enable --now fstrim.timer**

* REFLECTOR

**sudo pacman -S reflector && sudo reflector -c Brazil -a 5 --sort rate --save /etc/pacman.d/mirrorlist && sudo pacman -Syy sudo systemctl enable --now reflector.timer**

### SWAP FILE

Não criou partição de swap, porque [prefere](https://translate.googleusercontent.com/translate_c?depth=1&hl=pt-BR&ie=UTF8&langpair=auto|pt&pto=aue&rurl=translate.google.com&sp=nmt4&tbb=1&u=https://averagelinuxuser.com/linux-swap/&usg=ALkJrhiXonrwrzFm4EBtsQmfUxqZebXEhg) usar Swap File

1. Crie um arquivo de swap do tamanho(X) da sua RAM:

**dd if=/dev/zero of=/swapfile bs=1M count=512 status=progress**

As opções são:

**if**: O arquivo de entrada. Neste exemplo, estamos usando o **/dev/zero** qual fornecerá um fluxo de zero bytes.

**of**: O arquivo de saída. Vamos criar um arquivo no diretório raiz, chamado **swapfile**.

**bs**: o tamanho do bloco em bytes. Isso especifica quantos bytes ler do arquivo de entrada e gravar no arquivo de saída de cada vez.

**count**: quantos blocos ler e escrever. Multiplique este número pelo **bs** valor para obter o tamanho do arquivo.

1. Altere suas regras de acesso, formate e habilite:

**chmod 600 /swapfile && mkswap /swapfile && swapon /swapfile**

1. Aldicione a Swap File ao /etc/fstab:

**echo '/swapfile none swap defaults 0 0' >> /etc/fstab**

1. E verifique se o arquivo Swap está funcionando:

**free -m** ou **swapon --show** ou **swapon -s**

1. Para remover um arquivo swap, ele deve primeiro ser desligado:

**swapoff /swapfile && rm -f /swapfile**

* CACHE DE PACOTE DE LIMPEZA AUTOMÁTICA

Isso irá limpar o cache do pacote para manter apenas 1 versão após cada ação

**yay -S** **pacman-contrib**

**sudo mkdir** **/etc/pacman.d/hooks**

**sudo nano /etc/pacman.d/hooks/clean\_package\_cache.hook**

**[Trigger]**

**Operation = Upgrade**

**Operation = Install**

**Operation = Remove**

**Type = Package**

**Target = \***

**[Action]**

**Description = Cleaning pacman cache...**

**When = PostTransaction**

**Exec =** **/usr/bin/paccache –rk1 && /usr/bin/paccache –rk**

**paccache -ruk0 && paccache -rk1**

* REPOSITÓRIO MULTILIB

1. Habilitando

**sed -i '/multilib\]/,+1 s/^#//' /etc/pacman.conf && pacman -Sy**

**OU pelo nano /etc/pacman.conf , d**escomentando essas linhas:

**[multilib]**

**Include = /etc/pacman.d/mirrorlist**

1. **Removendo Pacotes Multilib,** que foram instalados do ***multilib***:

**pacman -R $(comm -12 <(pacman -Qq | sort) <(pacman -Slq multilib | sort))**

* **AUR (Arch User Repository)**

1. Inserir o repo AUR em **pacman.conf**:

**echo -e "[archlinuxfr]\nSigLevel = Never\nServer=http://repo.archlinux.fr/\$arch" >> /etc/pacman.conf**

1. Depois verifique se tudo está OK com o comando **cat**:

**cat /etc/pacman.conf**

1. Depois atualize o seu sistema pelo seguinte comando:

**pacman –Sy**

* PAMAC (Gerenciador de pacotes GUI)

Instalação via interface gráfica dos programas disponíveis nos repositórios oficiais do **Arch** Linux, bem como do **AUR** **(Arch User Repository)**.

1. Dependências necessárias

**sudo pacman -S** **--needed base-devel git wget yajl**

A opção **--needed** serve para ignorar pacotes que você já tem instalados, em vez de reinstalá-los.

1. O próximo passo é instalar o **YAY** (Compilando um pacote do AUR, ou pode ser instalado também via repositório)

**cd /tmp && git clone** [**https://aur.archlinux.org/yay.git**](https://aur.archlinux.org/yay.git) **&& cd yay && makepkg -si**

**# makepkg -sri --skippgpcheck(força instalação)**

A opção -**s**, resolve automaticamente e instala quaisquer dependências com o pacman antes de compilar, ele usa o pacman para isso.

A opção **-i** instala o pacote se ele foi compilado com sucesso.

1. Uma vez que **YAY** esteja instalado, agora vamos instalar o **Pamac**:

**yay -S pamac-aur**

* KVM\_QEMU\_VIRT-MANAGER

**sudo pacman –S virt-manager virt-viewer qemu qemu-arch-extra libvirt ovmf vde2 ebtables dnsmasq bridge-utils openbsd-netcat libguestfs**

**sudo systemctl enable libvirtd.service**

**sudo systemctl enable dnsmasq.service**

**sudo nano /etc/libvirt/libvirtd.conf**

**unix\_sock\_group = "libvirt"**

**unix\_sock\_rw\_perms = "0770"**

**sudo gpasswd -a $USER libvirt**

**sudo virsh net-list --all**

**sudo virsh net-autostart --network default**

* ADB

**pacman -S android-tools android-udev**

**usermod -aG** **adbusers $USER**

* ZSH (Z Shell) e FISH

**pacman -S zsh zsh-completions**

**chsh -s /usr/bin/zsh** **$USER**

**pacman -S fish**

**chsh -s /usr/bin/fish $USER**

**POSSÍVEIS CONFIGURAÇÕES**

Configuração Arquivos

Locale /etc/locale.conf /etc/locale.gen

Console fonts and Keymap /etc/vconsole.conf

Timezone /etc/timezone /etc/localtime

Hardware clock /etc/adjtime

Hostname /etc/hostname /etc/hosts

Lista branca (modules) /etc/modules-load.d/

Lista negra (modules) /etc/modprobe.d/

* **ALGUNS ATALHOS PARA NANO**

**CTRL+ W <PALAVRA> <ENTER>- procurar por string. palavras Ex. .br/**

**ALT+ \ e ALT+ / - Levar o cursor para o início e para o final do documento  
PageUp e PageDown - chave para subir e descer (pode utilizar as setas)**

**CTRL+V e CTR+Y - Rolar uma página para frente e para trás**

**CTRL+C - Mostrar a posição atual do cursor dentro do texto, incluindo linha, coluna e número do caractere**

**ALT+A - Ativar a marcação de textos (modo de seleção), efetuar a seleção usando as setas de direção do teclado  
CTRL+K - Recortar a linha toda onde se encontra o cursor e guarda   
CTRL+U - Colar o conteúdo do buffer de transferência no local está o cursor:.  
CTRL+X - para sair, se foi modificado, irá perguntar se quer salvar, responda y ou n.**

* PROBLEMAS? CHROOT MODO LIVE DO ARCH (VIA USB OU CD/DVD)

Baixar a ISO: [https://www.archlinux-br.org/](https://www.archlinux-br.org/download/)[download](https://www.archlinux-br.org/download)[/](https://www.archlinux-br.org/download/)

Dê o Boot pelo Live Cd e monte as partições existentes:

* **mount /dev/sdaX /mnt**
* **mount /dev/sdaX /mnt/home**
* **mount /dev/sdaX /mnt/boot**
* **swapon /dev/sdaX**

**e etc....**

Confirme os pontos de montagem: **lsblk**

Entre no arch-chroot: **arch-chroot /mnt** ...e faça os procedimentos...

COMANDOS BÁSICOS DO PACMAN & YAY

* PACMAN

## **ATUALIZAR O SISTEMA**

sudo pacman -Sy = sincroniza os repositórios.

sudo pacman -Su = procura por atualização.

sudo pacman -Syu = sincroniza os repositórios/procura por atualização.

sudo pacman -Syy = sincroniza os repositórios do Manjaro Linux.

sudo pacman -Syyu = sincronização total/procura por atualização.

sudo pacman -Syu --ignoregroup pacote1 , pacote2… = sincroniza os repositórios/procura por atualização e ignora os grupos dos pacotes solicitados

* **INSTALANDO PACOTES**

sudo pacman -S pacote = instala um pacote.

sudo pacman -S pacote --nonconfirm = Instala o pacote sem precisar confirmar com “yes/no ,S/N”…

sudo pacman -Sdd = instala ignorando as dependências.

sudo pacman -Se pacote = instala apenas as dependências.

sudo pacman -Sg pacote = Instala pacotes pertencem a um determinado grupo

sudo pacman -S --overwrite \\* pacote = Forca a instalação de um pacote

sudo pacman -Syu $(pacman -Qnq) = para atualizar tudo

sudo pacman -Sw pacote = apenas baixa o pacote e não o instala.

Pacman irá baixar o pacote como arquivo **<package\_name> .tar.xz** e armazená-lo no diretório **“/var/cache/pacman/pkg”**. Na verdade, todos os pacotes do Arch Linux estão disponíveis no formato **.TAR.XZ**

Todos os pacotes que você baixar com o pacman sempre estarão lá. Verifique quais pacotes estão lá.

**ls /var/cache/pacman/pkg**

Precisa descobrir pacotes específicos? Use “grep” para pesquisar um pacote específico.

**ls /var/cache/pacman/pkg | grep < package\_name >**

sudo pacman -A pacote.pkg.tar.gz = instala um pacote local.

sudo pacman -U home/user/arquivo.tar.xz = instalar pacotes baixados no pc

sudo pacman -U http://www.site.com/arquivo.tar.xz = instalar pacotes baixados via download

yes | sudo pacman -Syu = responde sim

* **REMOVENDO PACOTES**

sudo pacman -R pacote = remove um pacote.

sudo pacman -Rs pacote = remove o pacote junto com as dependências não usadas por outros pacotes.

sudo pacman -Rsn pacote = remove o pacote junto com as dependências não usadas por outros pacotes e junto com os arquivos de configuração.

sudo pacman -Rscn = Desinstala pacotes e suas dependências e seus registros, tudo.

sudo pacman -Rns $(pacman -Qqdt) = apaga pacotes desnecessários, sem dependências(órfãos)

sudo pacman -Rdd = elimina um pacote ignorando as dependências.

* **LIMPANDO O CACHE DE PACOTES**

sudo pacman -Sc = deleta do cache todos os pacotes antigos que não estão instalados atualmente e a base de dados de sincronização não utilizada

sudo pacman -Scc = limpa o cache, removendo todos os pacotes existentes no /var/cache/pacman/pkg/.

* **CONSULTANDO BASE DE DADOS DOS PACOTES**

sudo pacman -Ss pacote = pesquisar por pacotes na base de dados, pesquisando nomes e descrições

sudo pacman -Si pacote = mostra informações de um pacote não instalado.

Este método não possui realce de cor. Vadio! Não se preocupe; pacman já sabe como colorir a saída, mas a função está desabilitada por padrão. Quer ver a saída colorida? Edite o “pacman.conf” e descomente a opção “Color”.

**sudo nano /etc/pacman.conf**

sudo pacman -Qi pacote = mostra informações do pacote já instalado.

sudo pacman -Ql pacote = mostra todos os arquivos pertencentes ao pacote.

sudo pacman -Qo arquivo = mostra a qual pacote aquele arquivo pertence.

sudo pacman -Qu = mostra os pacotes que serão atualizados.

sudo pacman -Q = lista todos os pacotes instalados.

sudo pacman -Q | grep pacote = Se você está procurando por algum pacote específico, use “grep”.

sudo pacman -Qdt = lista pacotes desnecessários, sem dependências (órfãos)

sudo pacman -Qet = lista pacotes explicitamente instalados e que não são necessários como dependências

sudo pacman -Qem = lista pacotes instalados do repo AUR

sudo pacman -Qs pacote = procurar os pacotes já instalados

* **COMANDOS ADICIONAIS**

sudo pactree pacote = visualizar a árvore de dependência de um pacote

sudo pacman-optimize = otimiza a base de dados do pacman.

sudo pacman-mirrors.conf = para gerenciar pacman.cof

sudo pacman-mirrors -g = para gerar um novo mirrorlist

* YAY

yay -S <package-name> = Instalar um pacote ou aplicativo no AUR

yay -Ss <package-name> = Encontrar uma aplicação nos repôs. oficiais e AUR de cada vez

yay -Si <package-name> = Conhecer as informações de um determinado pacote

yay -U ruta-del-paquete = Instalar um pacote local

yay <package-name> =Apenas o nome do pacote e ele fará uma pesquisa

yay -Pu = Quais atualizações temos disponíveis

yay -Sy = Sincronizar apenas os pacotes do banco de dados

yay -Syu = Executar uma atualização do sistema

yay -Syua = Atualizar o sistema, incluindo os pacotes instalados do AUR

yay -S --noconfirm <package-name> = Instalar qualquer pacote sem confirmações

yay -Yc = Remover dependências indesejadas

yay -Scc = Limpar o cache dos aplicativos

yay -R <package-name> = Remover "apenas" um pacote ou aplicativo

yay -Rs <package-name> = Remover um pacote ou aplicativo junto com suas dependências

yay -Rnsc <nome do pacote> = Remover um pacote, suas dependências e configurações

man yay = Saber um pouco mais sobre o uso do yay, consulte o manual

