

Rocky Linux 10 Desktop — Pós-instalação completo (workstation)

Resumo

Guia prático para deixar o Rocky 10 Desktop pronto para trabalho: repositórios (EPEL/RPM Fusion/Flathub/ELRepo), GNOME e extensões, codecs multimídia, fontes, impressão/scan, Bluetooth, energia em notebooks, **drivers NVIDIA (método ELRepo ou RPM Fusion)**, rede/firewall/Cockpit, aplicativos essenciais (office, comunicação, mídia, dev GUI), backup/sincronização, segurança/hardening, sanity check, FAQ e checklist. Inclui **o que cada pacote faz**.

Sumário

- [1. Atualização e DNF turbo](#)
- [2. Repositórios: EPEL, RPM Fusion, Flathub, ELRepo](#)
- [3. Essenciais de terminal e arquivos \(com descrição\)](#)
- [4. GNOME ajustado \(Tweaks, extensões, escala\)](#)
- [5. Multimídia e codecs \(FFmpeg/GStreamer/VLC\)](#)
- [6. Fontes e idiomas](#)
- [7. Impressoras, scanners e Bluetooth](#)
- [8. Energia e thermals \(notebooks\)](#)
- [9. Drivers NVIDIA \(ELRepo ou RPM Fusion\)](#)
- [10. Rede, firewall e Cockpit](#)
- [11. Aplicativos \(produtividade, comunicação, mídia, dev GUI\)](#)
- [12. Backup pessoal e sincronização](#)
- [13. Segurança e hardening \(SELinux, updates, auditoria\)](#)
- [14. Verificações pós-reboot \(sanity check\)](#)
- [15. FAQ](#)
- [16. Checklist final](#)
- [17. Referências rápidas \(comandos\)](#)
- [Assinatura](#)

1. Atualização e DNF turbo

```
sudo dnf upgrade -y
sudo dnf install -y dnf-plugins-core
sudo dnf config-manager --set-enabled crb
sudo vim /etc/dnf/dnf.conf
```

Adicione/garanta:

```
fastestmirror=True
max_parallel_downloads=10
```

```
defaultyes=True
installonly_limit=3
```

2. Repositórios: EPEL, RPM Fusion, Flathub, ELRepo

```
# EPEL: repositório extra mantido pela comunidade para Enterprise Linux
sudo dnf install -y epel-release

# RPM Fusion: pacotes multimídia/driver/extra (free e nonfree)
sudo dnf install -y \
    https://download1.rpmfusion.org/free/el/rpmfusion-free-release-$(rpm -E
%rhel).noarch.rpm \
    https://download1.rpmfusion.org/nonfree/el/rpmfusion-nonfree-release-$(rpm -E
%rhel).noarch.rpm

# Flathub: loja de Flatpaks (apps desktop atualizados em sandbox)
sudo dnf install -y flatpak
flatpak remote-add --if-not-exists flathub
https://flathub.org/repo/flathub.flatpakrepo

# ELRepo: drivers de hardware (ex.: NVIDIA kmod), kernels e firmwares
sudo dnf install -y https://www.elrepo.org/elrepo-release-latest.el$(rpm -E
%rhel).noarch.rpm
```

3. Essenciais de terminal e arquivos (com descrição)

Instalação única dos básicos

```
sudo dnf install -y linux-firmware intel-gpu-firmware vim-enhanced bash-
completion git curl wget rsync htop btop fzf ripgrep jq yq tree unzip p7zip
p7zip-pluginsunar neovim tmux lsof strace bind-utils nmap traceroute telnet mtr
xfsprogs btrfs-progs exfatprogs ntfs-3g util-linux-user inxi tar
policycoreutils-python-utils setools-console
```

O que cada pacote faz (resumo)

Pacote	Para que serve
vim-enhanced	Editor de texto avançado (modo texto) — seu padrão preferido.
bash-completion	Autocompletar no bash para comandos/opções.
git	Controle de versão.
curl/wget	Download/requests HTTP/S via CLI.

Pacote	Para que serve
<code>rsync</code>	Sincronização/backup incremental via rede/local.
<code>htop/btop</code>	Monitores interativos de processos/recursos.
<code>fzf</code>	Busca fuzzy para histórico/arquivos/command palette no terminal.
<code>ripgrep (rg)</code>	Grep ultrarrápido recursivo.
<code>jq/yq</code>	Manipular JSON/YAML na linha de comando.
<code>tree</code>	Árvores de diretórios em texto.
<code>unzip/p7zip*/unar</code>	Descompactadores (ZIP, 7z, rar, etc.).
<code>neovim</code>	Editor moderno (opcional ao Vim).
<code>tmux</code>	Multiplexador de terminal (sessões persistentes).
<code>lsof</code>	Lista arquivos abertos e sockets.
<code>strace</code>	Tracing de syscalls (diagnóstico fino).
<code>bind-utils</code>	Ferramentas DNS (<code>dig</code> , <code>nslookup</code> , etc.).
<code>nmap</code>	Scanner de rede/portas.
<code>traceroute/mtr</code>	Rota de pacotes e diagnóstico de latência/loss.
<code>telnet</code>	Testes rápidos de portas em texto puro.
<code>xfspgfs/btrfs-progs</code>	Ferramentas para XFS/Btrfs.
<code>exfatprogs</code>	Suporte a exFAT (pendrives/cartões SD).
<code>ntfs-3g</code>	Suporte a NTFS (leitura/escrita).
<code>util-linux-user</code>	Ferramentas de usuários (ex.: <code>chsh</code> , etc.).
<code>inxi</code>	Resumo detalhado de hardware e drivers.
<code>tar</code>	Empacotar/desempacotar arquivos <code>.tar/.tar.gz</code> .
<code>policycoreutils-python-utils</code>	Ferramentas SELinux (<code>semanage</code> , etc.).
<code>setools-console</code>	Análise de políticas SELinux (linha de comando).

Definir `vim` como editor padrão:

```
sudo update-alternatives --set editor /usr/bin/vim
```

Ativar `fzf` por padrão:

```
echo 'export FZF_DEFAULT_COMMAND="rg --files --hidden --follow --glob \".!
{.git,node_modules,.venv}\""'"' >> ~/.bashrc
source ~/.bashrc
```

4. GNOME ajustado (Tweaks, extensões, escala)

```
sudo dnf install -y gnome-tweaks gnome-extensions-app \
  gnome-shell-extension-appindicator gnome-shell-extension-dash-to-dock
```

Pacote	Para que serve
<code>gnome-tweaks</code>	Ajustes finos de GNOME (temas, fontes, atalhos).
<code>gnome-extensions-app</code>	Gerenciar extensões do GNOME.
<code>...-appindicator</code>	Mostra ícones de apps na barra superior (tray).
<code>...-dash-to-dock</code>	Dock configurável (tamanho, auto-esconder, etc.).

Escala fracionária (monitores HiDPI):

```
gsettings set org.gnome.mutter experimental-features "['scale-monitor-
framebuffer']"
```

Backup de configurações:

```
dconf dump / > ~/dconf-backup.ini
# restore: dconf load / < ~/dconf-backup.ini
```

5. Multimídia e codecs (FFmpeg/GStreamer/VLC)

```
sudo dnf install -y ffmpeg vlc \
  gstreamer1-plugins-base gstreamer1-plugins-good gstreamer1-plugins-ugly \
  gstreamer1-plugins-bad-free gstreamer1-plugins-bad-freeworld gstreamer1-libav \
  libva-utils mesa-dri-drivers mesa-vulkan-drivers vulkan vulkan-tools
```

Pacote	Para que serve
<code>ffmpeg</code>	Conversão/reprodução/encapsulamento de áudio/vídeo.

Pacote	Para que serve
<code>vlc</code>	Player multimídia completo.
<code>gststreamer1-plugins-*</code>	Codecs e filtros (base/good/ugly/bad/libav).
<code>libva-utils</code>	Testes VA-API (aceleração de vídeo).
<code>mesa-*/vulkan*</code>	Drivers de vídeo abertos e suporte Vulkan.

Equalizador PipeWire (Flatpak):

```
flatpak install -y flathub com.github.wmm.easyeffects
```

6. Fontes e idiomas

```
sudo dnf install -y google-noto-sans-fonts google-noto-serif-fonts \
  google-noto-emoji-fonts fira-code-fonts jetbrains-mono-fonts \
  liberation-{mono,sans,serif}-fonts langpacks-pt langpacks-pt_BR \
  ibus-typing-booster
```

Pacote	Para que serve
<code>google-noto-*</code>	Conjunto amplo de fontes e emojis.
<code>fira-code-fonts/jetbrains-mono-fonts</code>	Fontes com ligaduras para código.
<code>liberation-*</code>	Substitutas compatíveis às Microsoft fonts.
<code>langpacks-pt/pt_BR</code>	Pacotes de idioma e localização PT-BR.
<code>ibus-typing-booster</code>	Sugestões de digitação e correções.

Microsoft Core Fonts (opcional):

```
sudo dnf install -y cabextract
sudo dnf install -y
https://downloads.sourceforge.net/project/mscorefonts2/rpms/msttcore-fonts-
installer-2.6-1.noarch.rpm
```

7. Impressoras, scanners e Bluetooth

```
sudo dnf install -y cups system-config-printer hplip
sudo systemctl enable --now cups
```

```
sudo dnf install -y simple-scan sane-backends

sudo dnf install -y blueman bluez bluez-tools
sudo systemctl enable --now bluetooth
```

Pacote	Para que serve
<code>cups</code>	Servidor de impressão.
<code>system-config-printer</code>	GUI para adicionar impressoras.
<code>hplip</code>	Suporte HP (drivers/firmware utilitários).
<code>simple-scan/sane-backends</code>	Digitalização de documentos.
<code>blueman/bluez*</code>	Stack e GUI para Bluetooth.

8. Energia e thermals (notebooks)

```
sudo dnf install -y tlp thermald powertop
sudo systemctl enable --now tlp thermald
sudo powertop --auto-tune
```

Pacote	Para que serve
<code>tlp</code>	Perfis de economia de energia (CPU, discos, PCI).
<code>thermald</code>	Controla throttling térmico da CPU Intel.
<code>powertop</code>	Diagnóstico e auto-tuning de consumo.

9. Drivers NVIDIA (ELRepo **ou** RPM Fusion)

Escolha um método e não misture. Se seu equipamento não tiver GPU NVIDIA, ignore esta seção.

9.1 Pré-checagens

```
# Detectar GPU NVIDIA
lspci | grep -i -E 'nvidia|vga'

# Kernel e headers (úteis para akmods)
uname -r
rpm -q kernel-headers kernel-devel
```

9.2 Observação sobre Secure Boot

- Verifique o estado:

```
sudo mokutil --sb-state
```

- Se **Secure Boot** estiver habilitado:
 - **RPM Fusion (akmod)**: importe a chave local do akmods no MOK para permitir o carregamento do módulo assinado:

```
sudo mokutil --import /etc/pki/akmods/certs/akmods.pem  
# defina uma senha; no reboot, confirme no MokManager
```

- **Alternativa**: desabilite Secure Boot no firmware (BIOS/UEFI) se a política da máquina permitir.

9.3 Método A — **ELRepo (kmod-nvidia)**

Vantagem: **kmod** já pré-compilado para o kernel do EL, tende a ser estável.

```
# Certifique-se de ter o ELRepo habilitado (seção 2)  
sudo dnf --enablerepo=elrepo-kernel install -y nvidia-detect || true  
nvidia-detect || true # sugere a série do driver  
  
# Driver principal e utilitários  
sudo dnf --enablerepo=elrepo-kernel install -y \  
    kmod-nvidia nvidia-x11-drv nvidia-settings nvidia-modprobe  
  
# (Opcional) CUDA/OpenCL userspace do ELRepo (se disponível)  
# sudo dnf --enablerepo=elrepo-kernel install -y xorg-x11-drv-nvidia-cuda
```

Reinicie:

```
sudo reboot
```

Validação:

```
nvidia-smi  
inxi -Gxx
```

9.4 Método B — **RPM Fusion (akmod-nvidia)**

Vantagem: **akmod** compila módulo para seu kernel atual (requer toolchain/headers).

```
# Garantir toolchain para build do akmod  
sudo dnf install -y kernel-headers kernel-devel gcc make
```

```
# Driver NVIDIA via RPM Fusion
sudo dnf install -y akmod-nvidia xorg-x11-drv-nvidia-cuda
# (Opcional) pacotes NVENC/NVDEC/Vulkan extra podem ser instalados conforme
necessidade
```

Reinicie:

```
sudo reboot
```

Validação:

```
nvidia-smi
inxi -Gxx
```

Notas

- Os pacotes tratam o **blacklist do Nouveau** automaticamente. Se necessário, faça manualmente:

```
echo -e "blacklist nouveau\noptions nouveau modeset=0" | sudo tee
/etc/modprobe.d/blacklist-nouveau.conf
sudo dracut --force
```

- Para tearing menor em Xorg: habilite DRM KMS completo (às vezes já vem por padrão):

```
echo "options nvidia_drm modeset=1" | sudo tee /etc/modprobe.d/nvidia-drm-
modeset.conf
```

Reinicie após.

10. Rede, firewall e Cockpit

```
# SSH server (opcional)
sudo systemctl enable --now sshd
sudo firewall-cmd --permanent --add-service=ssh
sudo firewall-cmd --reload

# Cockpit: administração web do sistema
sudo dnf install -y cockpit
sudo systemctl enable --now cockpit.socket
```



```
sudo firewall-cmd --permanent --add-service=cockpit
sudo firewall-cmd --reload
```

Pacote/Serviço	Para que serve
<code>sshd</code>	Acesso remoto seguro por SSH.
<code>cockpit</code>	Painel web para gerenciar serviços/updates/armazenamento.

Exemplos de liberação:

```
sudo firewall-cmd --permanent --add-service=http
sudo firewall-cmd --permanent --add-service=https
sudo firewall-cmd --permanent --add-port=3000/tcp
sudo firewall-cmd --reload
```

11. Aplicativos (produtividade, comunicação, mídia, dev GUI)

11.1 Produtividade/Office

```
sudo dnf install -y libreoffice libreoffice-langpack-pt-BR \
    okular evince gnome-sushi file-roller
```

Pacote	Para que serve
<code>libreoffice + langpack-pt-BR</code>	Suite office com PT-BR.
<code>okular/evince</code>	Leitores de PDF/PS.
<code>gnome-sushi</code>	Prévia de arquivos no Nautilus (barra de espaço).
<code>file-roller</code>	Compactador/gerenciador de arquivos (GUI).

11.2 Comunicação (Flatpak)

```
flatpak install -y flathub com.discordapp.Discord \
    org.telegram.desktop com.slack.Slack com.skype.Client us.zoom.Zoom \
    com.github.eneshecan.WhatsAppForLinux
```

App	Para que serve
Discord/Telegram/Slack/Skype/Zoom/WhatsApp	Mensageria e reuniões.

11.3 Navegadores

```
sudo dnf install -y chromium
sudo dnf install -y https://dl.google.com/linux/direct/google-chrome-
stable_current_x86_64.rpm
# Alternativas:
# flatpak install -y flathub org.mozilla.firefox
# flatpak install -y flathub com.brave.Browser
```

11.4 Mídia/Criação/Streaming

```
sudo dnf install -y gimp inkscape darktable shotwell obs-studio
flatpak install -y flathub com.obsproject.Studio \
  com.spotify.Client org.audacityteam.Audacity
```

App	Para que serve
<code>gimp</code>	Edição de imagens (raster).
<code>inkscape</code>	Vetorial (SVG).
<code>darktable</code>	Fluxo RAW/fotografia.
<code>shotwell</code>	Gerenciador de fotos.
<code>obs-studio</code>	Gravação/stream de tela e vídeo.
<code>Spotify/Audacity</code>	Música/edição de áudio.

11.5 Dev GUI (DB/HTTP)

```
flatpak install -y flathub io.dbeaver.DBeaverCommunity \
  com.getpostman.Postman com.konghq.insomnia
```

App	Para que serve
DBeaver	Client universal de bancos de dados.
Postman/Insomnia	Testes de APIs REST/GraphQL.

12. Backup pessoal e sincronização

```
sudo dnf install -y deja-dup borgbackup restic rclone
```

Pacote	Para que serve
<code>deja-dup</code>	Backup gráfico (integração GNOME).

Pacote	Para que serve
<code>borgbackup</code>	Backup deduplicado, eficiente e verificável.
<code>restic</code>	Backup rápido com criptografia.
<code>rclone</code>	Sincronização com clouds (S3, GDrive, etc.).

13. Segurança e hardening (SELinux, updates, auditoria)

13.1 SELinux (preferido: **Enforcing**)

```
getenforce
sudo setenforce 0    # permissivo temporário, diagnóstico
sudo vim /etc/selinux/config
# SELINUX=enforcing  (recomendado)
```

13.2 Updates automáticos

```
sudo dnf install -y dnf-automatic
sudo vim /etc/dnf/automatic.conf
# apply_updates = yes
sudo systemctl enable --now dnf-automatic.timer
```

13.3 Auditoria e antivírus (opcional)

```
sudo dnf install -y lynis clamav clamav-update
sudo freshclam
sudo lynis audit system
```


14. Verificações pós-reboot (sanity check)

```
inxi -Gxx
pactl info
ip a && nmcli con show
ffmpeg -codecs | head
flatpak remotes
# NVIDIA (se instalado):
nvidia-smi
```

15. FAQ

NVIDIA: ELRepo (kmod) ou RPM Fusion (akmod)?

- **ELRepo/kmod**: driver pré-compilado para kernels EL, muito estável.
- **RPM Fusion/akmod**: compila módulo para seu kernel (requer headers/gcc). Não misture métodos.

Wayland x Xorg com NVIDIA? Wayland funciona nas versões recentes, mas alguns workflows ainda preferem Xorg por compatibilidade. Escolha na tela de login (ícone .

Flatpak ou RPM? Flatpak para apps desktop atualizados e isolados; RPM para libs/CLIs do sistema.

16. Checklist final

- ☐ Sistema atualizado e DNF otimizado
 - ☐ EPEL/RPM Fusion/Flathub/ELRepo habilitados
 - ☐ GNOME Tweaks + extensões (dock, appindicator)
 - ☐ Codecs instalados (FFmpeg/GStreamer/VLC)
 - ☐ Fontes PT-BR e programação (Fira Code/JetBrains Mono)
 - ☐ Impressão/scan e Bluetooth funcionando
 - ☐ TLP/ThermalD/PowerTop aplicados (notebooks)
 - ☐ **Driver NVIDIA** instalado e `nvidia-smi` OK (se aplicável)
 - ☐ Firewall/Cockpit configurados
 - ☐ Apps essenciais (office/comunicação/mídia/dev GUI)
 - ☐ Backup (Deja-Dup/Borg/Restic) e/ou sync (rclone)
 - ☐ SELinux em **enforcing** (preferido) e `dnf-automatic.timer` ativo
-

17. Referências rápidas (comandos)

Habilitar serviços no firewall:

```
sudo firewall-cmd --permanent --add-service={ssh,http,https,cockpit}
sudo firewall-cmd --reload
```

Abrir portas específicas:

```
sudo firewall-cmd --permanent --add-port=3000/tcp
sudo firewall-cmd --permanent --add-port=3001-3005/tcp
sudo firewall-cmd --reload
```

Verificar serviços:

```
systemctl status sshd
systemctl status bluetooth
systemctl status cups
```

Assinatura

Criado por **Jeferson Salles** LinkedIn: <https://www.linkedin.com/in/jmsalles/> E-mail: jefersonmattossalles@gmail.com.