

Création d'un kiosk linux sous Rockylinux

Un kiosque permet l'affichage d'un panneau numérique statique (par exemple, un panneau de menu dans un fast-food ou un affichage de l'état d'un vol dans un aéroport) ou interactif (par exemple, des directions dans un centre commercial, plus d'informations dans un musée). Cet article présente la création d'un kiosque sous Linux.

Tout d'abord, il faut commencer par une installation Linux, comme par exemple Rocky Linux 9.1, comme point de départ. Mais on peut également choisir d'autres distributions. Bien que les instructions ici fonctionnent bien avec Rocky Linux, il faudra peut-être y apporter des modifications mineures pour d'autres distributions.

Voici les tâches nécessaires :

- Installer et configurer le système graphique de base
- Créer un utilisateur de kiosque sans mot de passe.
- Configurer le gestionnaire de bureau Linux (GDM dans ce cas) pour connecter automatiquement l'utilisateur.
- Configurer l'utilisateur pour exécuter un profil « gnome-kiosk-script-session » restreint qui exécute un script shell pour lancer une application appropriée, par exemple Firefox avec certaines options de ligne de commande.

Installation et configuration du système graphique de base

Tout d'abord, il faut ajouter le système graphique de base que **dnf** permet d'installer facilement en utilisant le groupe **base-x**:

```
dnf -q group info base-x
Group: base-x
Description: Local X.org display server
Mandatory Packages:
  glx-utils
  mesa-dri-drivers
  plymouth-system-theme
  xorg-x11-drv-evdev
  xorg-x11-drv-fbdev
  xorg-x11-drv-libinput
  xorg-x11-drv-vmware
  xorg-x11-drv-wacom
  xorg-x11-server-Xorg
  xorg-x11-utils
  xorg-x11-xauth
  xorg-x11-xinit
  xorg-x11-xinit-session
```

```
dnf group install base-x
```



Pour les matériels à base de chipset intel il faut activer le module i915: `modprobe i915`

Puis il faut installer quelques paquets, que toutes les distributions de la famille Red Hat proposent sans aucun dépôt supplémentaire, avec cette commande:

```
dnf install gnome-kiosk gnome-kiosk-script-session gdm firefox
```

Liste des paquets installés: ModemManager-glib accountsservice accountsservice-libs adwaita-cursor-theme adwaita-icon-theme alsa-lib at-spi2-atk at-spi2-core atk avahi-glib bluez-libs bluez-obexd bolt bubblewrap cheese-libs clutter clutter-gst3 clutter-gtk cogl color-filesystem colord colord-gtk colord-libs cups-libs cups-pk-helper dbus-daemon dbus-glib dbus-tools dconf enchant2 evolution-data-server evolution-data-server-langpacks exempi exiv2 exiv2-libs fdk-aac-free firefox flac-libs fuse fuse-common gcr gcr-base gdisk gdk-pixbuf2-modules gdm gedit geoclue2 geoclue2-libs geocode-glib giflib gjs gnome-autoar gnome-bluetooth gnome-bluetooth-libs gnome-color-manager gnome-control-center gnome-control-center-filesystem gnome-desktop3 gnome-keyring gnome-keyring-pam gnome-kiosk gnome-kiosk-script-session gnome-online-accounts gnome-remote-desktop gnome-session gnome-session-wayland-session gnome-session-xsession gnome-settings-daemon gnome-shell gnome-shell-extension-background-logo gnome-tour graphene gsm gsound gspell gstreamer1 gstreamer1-plugins-bad-free gstreamer1-plugins-base gstreamer1-plugins-good gstreamer1-plugins-good-gtk gtk-update-icon-cache gtk3 gtk4 gtksourceview4 gvfs gvfs-client hicolor-icon-theme highcontrast-icon-theme hunspell hunspell-en-US hunspell-filesystem ibus ibus-gtk3 ibus-libs ibus-setup iio-sensor-proxy iso-codes jbigkit-libs lame-libs lcms2 libXdamage libasyncns libatasmart libblockdev libblockdev-crypto libblockdev-fs libblockdev-loop libblockdev-mdraid libblockdev-part libblockdev-swap libblockdev-utils libbytesize libcanberra libcanberra-gtk3 libcdio libcdio-paranoia libdvdnav libdvdread libexif libgdata libgexiv2 libglvnd-gles libgnomekbd libgsf libgtop2 libgweather libgxps libhandy libical libical-glib libiptcdata libldac libnma libnotify libogg libosinfo libpeas libpeas-gtk libpeas-loader-python3 libsbcl libsecret libshout libsndfile libsrtp libtheora libtiff libtool-ltdl libtracker-sparql libudisks2 libv4l libvisual libvorbis libvpx libwayland-cursor libwayland-egl libwebp libxcvt libxkbcommon libxkbcommon-x11 libxklavier low-memory-monitor mesa-vulkan-drivers mobile-broadband-provider-info mozilla-filesystem mpg123-libs mutter nm-connection-editor openjpeg2 opus orc osinfo-db osinfo-db-tools pinentry pinentry-gnome3 pipewire pipewire-alsa pipewire-gstreamer pipewire-jack-audio-connection-kit pipewire-jack-audio-connection-kit-libs pipewire-libs pipewire-pulseaudio poppler poppler-data poppler-glib power-profiles-daemon pulseaudio-libs pulseaudio-libs-glib2 python3-cairo python3-gobject rest rocky-indexhtml rtkit setxkbmap sound-theme-freedesktop soundtouch speex startup-notification switcheroo-control taglib totem-pl-parser tracker tracker-miners twolame-libs udisks2 upower volume_key-libs vulkan-loader wavpack webrtc-audio-processing wireplumber wireplumber-libs xcb-util xdg-desktop-portal xdg-desktop-portal-gnome xdg-desktop-portal-gtk xdg-user-dirs xdg-user-dirs-gtk xorg-x11-server-Xwayland zenity



Le paquet **gnome-kiosk** fournit une variante légère du shell GNOME, tandis que le paquet **gdm** fournit le gestionnaire de bureau GNOME. Les deux sont suffisants pour se connecter et démarrer un shell simple.

Création d'un utilisateur

Ensuite, il faut créer un utilisateur qui exécutera le kiosque. Cet utilisateur ne doit pas avoir d'autorisations spéciales, ni de mot de passe.

```
# Créer un utilisateur
useradd --create-home kiosk

# Supprimer le mot de passe
passwd -d kiosk
```

Et c'est déjà tout. Enfin... presque. Il peut être judicieux de vérifier le service sshd et de s'assurer que les utilisateurs sans mot de passe ne peuvent pas se connecter.

```
# S'assurer que les mots de passe vides ne sont pas autorisés
sed -i 's/.*PermitEmptyPasswords/PermitEmptyPasswords no/g'
/etc/ssh/sshd_config

# Redémarrer sshd
systemctl restart sshd.service
```

Configurer l'utilisateur pour exécuter un profil

Il faut également besoin d'un peu de configuration, afin que tout fonctionne comme prévu. Tout d'abord, créer un fichier qui connectera automatiquement l'utilisateur. Ouvrir le fichier `/etc/gdm/custom.conf` et vérifier qu'il ressemble à ce qui suit.

```
[daemon]
AutomaticLoginEnable=True
AutomaticLogin=kiosk

[security]

[xdmcp]

[chooser]

[debug]
```

Ensuite, il faut configurer la session souhaitée. Cela peut être fait avec un autre fichier de configuration `/var/lib/AccountsService/users/kiosk`. Dans ce fichier, indiquer à GDM que la session **gnome-kiosk-script** doit être exécutée lorsque l'utilisateur du **kiosk** se connecte.

```
[User]
Session=gnome-kiosk-script
SystemAccount=false
```

Cette session nécessite un script à un emplacement spécifique. Pour le cas d'utilisation, on veut exécuter le navigateur **Firefox** installé ci-dessus, et appeler la page souhaitée. Pour ce faire, il faut créer un répertoire dans le répertoire d'accueil du kiosque:

```
mkdir -p /home/kiosk/.local/bin /home/kiosk/.config /home/kiosk/.cache
/home/kiosk/.mozilla
chown -R kiosk:kiosk /home/kiosk/
```

Créer également le script `/home/kiosk/.local/bin/gnome-kiosk-script`

```
#!/bin/sh
while true; do
  firefox -kiosk http://xx.xxx.xxx.xx:3000
  sleep 1
done
```

La boucle prend en charge les situations où un utilisateur peut avoir arrêté Firefox ou l'avoir planté. Ce script doit être exécutable. Il faut donc modifier les autorisations en conséquence.

```
# Rendre exécutable
chmod 0755 /home/kiosk/.local/bin/gnome-kiosk-script

# Garantir les autorisations correctes pour /home/kiosk
chown -R kiosk:kiosk /home/kiosk
```

Configuration du gestionnaire de bureau Linux

Il y a encore une chose dont il faut s'occuper. Lors du redémarrage, il se peut que le système ne démarre pas en mode graphique et, par conséquent, ne propose jamais le mode kiosque souhaité. Il faut donc encore deux commandes et pour s'assurer du démarrage en mode graphique.

```
# Activer gdm
systemctl enable gdm.service

# S'assurer de démarrer en mode graphique
systemctl set-default graphical.target
```

Redémarrer la machine et vous devriez être accueilli par l'image suivante.

From:

<http://www.ouarte.garden/> - **dwndoc**

Permanent link:

<http://www.ouarte.garden/doku.php?id=howto:kiosk-rocky-linux>

Last update: **2025/02/19 10:59**

