

GDM

Related articles

- [GNOME](#)
- [Display manager](#)

From [GDM - GNOME Display Manager](#): "The GNOME Display Manager (GDM) is a program that manages graphical display servers and handles graphical user logins."

[Display managers](#) provide [X Window System](#) and [Wayland](#) users with a graphical login prompt.

Contents

[hide]

- 1Installation
- 2Starting
 - 2.1Autostarting applications
- 3Configuration
 - 3.1Log-in screen background image
 - 3.2DConf configuration
 - 3.2.1Log-in screen logo
 - 3.2.2Changing the cursor theme
 - 3.2.3Larger font for log-in screen
 - 3.2.4Turning off the sound
 - 3.2.5Configure power button behavior
 - 3.2.6Enabling tap-to-click
 - 3.2.7Disable/Enable Accessibility Menu
 - 3.3Keyboard layout
 - 3.4Change the language
 - 3.5Users and login
 - 3.5.1Automatic login
 - 3.5.2Passwordless login
 - 3.5.3Passwordless shutdown for multiple sessions
 - 3.5.4Enable root login in GDM
 - 3.5.5Hide user from login list
 - 3.6Setup default monitor settings
 - 3.7Configure X server access permission
- 4Troubleshooting
 - 4.1Failure to use proprietary NVIDIA driver
 - 4.2Failure on logout
 - 4.3Rootless Xorg
 - 4.4Use Xorg backend
 - 4.5Incomplete removal of gdm
- 5See also

Installation

GDM can be [installed](#) with the [gdm](#) package, and it is installed as part of the [gnome](#) group.

If you would prefer to use legacy GDM which was used in GNOME 2 and has its own configuration utility, install the [gdm-01d](#)^{AUR} package. Note that the rest of this article discusses current GDM, not legacy GDM, unless indicated otherwise.

You might also wish to install the following:

- **gdm3setup** — An interface to configure GDM3, autologin options and change Shell theme
<https://github.com/Nano77/gdm3setup> || [gdm3setup-utils](#)^{AUR}

Starting

To start GDM at boot time [enable](#) `gdm.service`.

Autostarting applications

One might want to autostart certain commands, such as *xrandr* for instance, on login. This can be achieved by adding a command or script to a location that is sourced by the display manager. See [Display manager#Autostarting](#) for a list of supported locations.

Note: The `/etc/gdm/Init` directory is no longer a supported location, see [\[1\]](#).

Configuration

Log-in screen background image

Note:

- Since GNOME 3.16, GNOME Shell themes are now stored as binary files (gresource).
- This change will be overwritten on subsequent updates of [gnome-shell](#).

Firstly, you need to extract the existing GNOME Shell theme to a folder in your home directory. You can do this using the following script:

```
extractgst.sh

#!/bin/sh
gst=/usr/share/gnome-shell/gnome-shell-theme.gresource
workdir=${HOME}/shell-theme

for r in `gresource list $gst`; do
    r=${r#\org/gnome/shell/}
    if [ ! -d $workdir/${r%/*} ]; then
        mkdir -p $workdir/${r%/*}
    fi
done

for r in `gresource list $gst`; do
    gresource extract $gst $r >$workdir/${r#\org/gnome/shell/}
done
```

Navigate to the created directory. You should find that the theme files have been extracted to it. Now copy your preferred background image to this directory.

Next, you need to create a file in the directory with the following content:

```
gnome-shell-theme.gresource.xml

<?xml version="1.0" encoding="UTF-8"?>
<gresources>
  <gresource prefix="/org/gnome/shell/theme">
    <file>calendar-arrow-left.svg</file>
    <file>calendar-arrow-right.svg</file>
    <file>calendar-today.svg</file>
```

```

<file>checkbox-focused.svg</file>
<file>checkbox-off-focused.svg</file>
<file>checkbox-off.svg</file>
<file>checkbox.svg</file>
<file>close-window-active.svg</file>
<file>close-window-hover.svg</file>
<file>close-window.svg</file>
<file>close.svg</file>
<file>corner-ripple-ltr.png</file>
<file>corner-ripple-rtl.png</file>
<file>dash-placeholder.svg</file>
<file>filter-selected-ltr.svg</file>
<file>filter-selected-rtl.svg</file>
<file>gnome-shell-high-contrast.css</file>
<file>gnome-shell.css</file>
<file>icons/message-indicator-symbolic.svg</file>
<file>logged-in-indicator.svg</file>
<file>no-events.svg</file>
<file>no-notifications.svg</file>
<file>filename</file>
<file>pad-osd.css</file>
<file>page-indicator-active.svg</file>
<file>page-indicator-checked.svg</file>
<file>page-indicator-hover.svg</file>
<file>page-indicator-inactive.svg</file>
<file>process-working.svg</file>
<file>running-indicator.svg</file>
<file>source-button-border.svg</file>
<file>summary-counter.svg</file>
<file>toggle-off-hc.svg</file>
<file>toggle-off-intl.svg</file>
<file>toggle-off-us.svg</file>
<file>toggle-on-hc.svg</file>
<file>toggle-on-intl.svg</file>
<file>toggle-on-us.svg</file>
<file>ws-switch-arrow-down.png</file>
<file>ws-switch-arrow-up.png</file>
</gresource>
</gresources>

```

Replace **filename** with the filename of your background image.

Now, open the `gnome-shell.css` file in the directory and change the `#lockDialogGroup` definition as follows:

```
#lockDialogGroup {
```

```
background: #2e3436 url(filename);
background-size: [WIDTH]px [HEIGHT]px;
background-repeat: no-repeat;
}
```

Set `background-size` to the resolution that GDM uses, this might not necessarily be the resolution of the image. For a list of display resolutions see [Display resolution](#). Again, set **filename** to be the name of the background image.

Finally, compile the theme using the following command:

```
$ glib-compile-resources gnome-shell-theme.gresource.xml
```

Then copy the resulting `gnome-shell-theme.gresource` file to the `/usr/share/gnome-shell` directory.

Then restart `gdm.service` (note that simply logging out is not enough) and you should find that it is using your preferred background image.

For more information, please see the following [forum thread](#).

DConf configuration

Some GDM settings are stored in a DConf database. They can be configured either by adding *keyfiles* to the `/etc/dconf/db/gdm.d` directory and then recompiling the GDM database by running `dconf update` as root or by logging into the GDM user on the system and changing the setting directly using the `gsettings` command line tool. Note that for the former approach, a GDM profile file is required - this must be created manually as it is no longer shipped upstream, see below:

```
/etc/dconf/profile/gdm

user-db:user
system-db:gdm
file-db:/usr/share/gdm/greeter-dconf-defaults
```

For the latter approach, you can log into the GDM user with the command below:

```
# machinectl shell gdm@
```

Log-in screen logo

Either create the following keyfile

```
/etc/dconf/db/gdm.d/02-logo

[org/gnome/login-screen]
logo='/path/to/logo.png'
```

and then recompile the GDM database or alternatively log in to the GDM user and execute the following:

```
$ gsettings set org.gnome.login-screen logo '/path/to/logo.png'
```

Changing the cursor theme

GDM disregards [GNOME](#) cursor theme settings and it also ignores the cursor theme set according to the [XDG specification](#). To change the cursor theme used in GDM, either create the following keyfile

```
/etc/dconf/db/gdm.d/10-cursor-settings

[org/gnome/desktop/interface]
cursor-theme='theme-name'
```

and then recompile the GDM database or alternatively log in to the GDM user and execute the following:

```
$ gsettings set org.gnome.desktop.interface cursor-theme 'theme-name'
```

Larger font for log-in screen

Click on the accessibility icon at the top right of the screen (a white circle with the silhouette of a person in the centre) and check the *Large Text* option.

To set a specific scaling factor, you can create the following keyfile:

```
/etc/dconf/db/gdm.d/03-scaling

[org/gnome/desktop/interface]
text-scaling-factor='1.25'
```

and then recompile the GDM database or alternatively log in to the GDM user and execute the following:

```
$ gsettings set org.gnome.desktop.interface text-scaling-factor '1.25'
```

Turning off the sound

This tweak disables the audible feedback heard when the system volume is adjusted (via keyboard) on the login screen.

Either create the following keyfile:

```
/etc/dconf/db/gdm.d/04-sound

[org/gnome/desktop/sound]
event-sounds='false'
```

and then recompile the GDM database or alternatively log in to the GDM user and execute the following:

```
$ gsettings set org.gnome.desktop.sound event-sounds 'false'
```

Configure power button behavior

Note:

- The [logind settings](#) for the power button are overridden by GNOME Settings Daemon. [\[2\]](#)
- As of October 2015, the power button cannot be set to *interactive*. [\[3\]](#)
- In some cases, this setting will be ignored and hardcoded defaults will be used. [\[4\]](#)

Warning: Please note that the [acpid](#) daemon also handles the "power button" and "hibernate button" events. Running both systems at the same time may lead to unexpected behaviour.

Either create the following keyfile:

```
/etc/dconf/db/gdm.d/05-power

[org/gnome/settings-daemon/plugins/power]
power-button-action='action'
```

and then recompile the GDM database or alternatively log in to the GDM user and execute the following:

```
$ gsettings set org.gnome.settings-daemon.plugins.power power-button-
action 'action'
```

where *action* can be one of `nothing`, `suspend` or `hibernate`.

Enabling tap-to-click

Tap-to-click is disabled in GDM (and GNOME) by default, but you can easily enable it with a dconf setting.

Note: If you want to do this under X, you have to first set up correct X server access permissions - see [#Configure X server access permission](#).

To directly enable tap-to-click, use:

```
# sudo -u gdm gsettings set org.gnome.desktop.peripherals.touchpad tap-to-
click true
```

If you prefer to do this with a GUI, use:

```
# sudo -u gdm dconf-editor
```

To check the if it was set correctly, use:

```
$ sudo -u gdm gsettings get org.gnome.desktop.peripherals.touchpad tap-to-
click
```

If you get the error `dconf-WARNING **: failed to commit changes to dconf: Error spawning command line`, make sure dbus is running:

```
$ sudo -u gdm dbus-launch gsettings set
org.gnome.desktop.peripherals.touchpad tap-to-click true
```

Disable/Enable Accessibility Menu

To disable or enable the Accessibility Menu, set the following key in dconf editor:

```
# machinectl shell gdm@
# gsettings set org.gnome.desktop.interface toolkit-accessibility false
# exit
```

The menu is disabled when the key is false, enabled when it is true.

Keyboard layout

See [Keyboard configuration in Xorg#Using X configuration files](#).

Tip: See [Wikipedia:ISO 3166-1](#) for a list of keymaps.

Alternatively, if the package [gnome-control-center](#) is installed, the keyboard layout(s) can be configured using a graphical frontend. Start *gnome-control-center* and navigate to *Region & Language* -> *Input Sources*. Then, in the header bar, hit the *Login Screen* toggle button to configure the keyboard layout for GDM specifically.

Users of GDM 2.x (legacy GDM) may need to edit `~/.dmrc` as shown below:

```
~/.dmrc
-----
[Desktop]
Language=de_DE.UTF-8    # change to your default lang
Layout=de      nodeadkeys # change to your keyboard layout
```

Change the language

To change the GDM language, ensure that [gnome-control-center](#) is installed. Then, start *gnome-control-center* and choose *Region & Language*. In the header bar, check the *Login Screen* toggle button. Finally, click on *Language* and choose your language from the list. You will be prompted for your root password.

Tip: By adding 2 different input languages, logging out then selecting your default language GDM will remember your choice once the second option is removed.

Users and login

Automatic login

To enable automatic login with GDM, add the following to `/etc/gdm/custom.conf` (replace *username* with your own):

```
/etc/gdm/custom.conf
-----
# Enable automatic login for user
[daemon]
AutomaticLogin=username
AutomaticLoginEnable=True
```

Tip: If GDM fails after adding these lines, comment them out from a TTY.

or for an automatic login with a delay:

```
/etc/gdm/custom.conf
```

```
[daemon]

TimedLoginEnable=true
TimedLogin=username
TimedLoginDelay=1
```

You can set the session used for automatic login (replace `gnome-xorg` with desired session):

```
/var/lib/AccountsService/users/username

XSession=gnome-xorg
```

Passwordless login

If you want to bypass the password prompt in GDM then simply add the following line on the first line of `/etc/pam.d/gdm-password`:

```
auth sufficient pam_succeed_if.so user ingroup nopasswdlogin
```

Then, add the group `nopasswdlogin` to your system. See [Groups](#) for group descriptions and group management commands.

Now, add your user to the `nopasswdlogin` group and you will only have to click on your username to login.

Warning:

- Do **not** do this for a **root** account.
- You won't be able to change your session type at login with GDM anymore. If you want to change your default session type, you will first need to remove your user from the `nopasswdlogin` group.

Passwordless shutdown for multiple sessions

GDM uses polkit and logind to gain permissions for shutdown. You can shutdown the system when multiple users are logged in by setting:

```
/etc/polkit-1/localauthority.conf.d/org.freedesktop.logind.policy

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE policyconfig PUBLIC
"-//freedesktop//DTD PolicyKit Policy Configuration 1.0//EN"
"http://www.freedesktop.org/standards/PolicyKit/1.0/policyconfig.dtd
```



```

    <message>System policy prevents shutting down the system when other
    users are logged in</message>
    <defaults>
        <allow_inactive>yes</allow_inactive>
        <allow_active>yes</allow_active>
    </defaults>
</action>

</policyconfig>

```

You can find all available logind options (e.g. reboot-multiple-sessions) [here](#).

Enable root login in GDM

It is not advised to login as root, but if necessary you can edit `/etc/pam.d/gdm-password` and add the following line before the line `auth required pam_deny.so`:

```
/etc/pam.d/gdm-password
```

```
auth                sufficient      pam_succeed_if.so uid eq 0 quiet
```

The file should look something like this:

```
/etc/pam.d/gdm-password
```

```

...
auth                sufficient      pam_succeed_if.so uid eq 0 quiet
auth                sufficient      pam_succeed_if.so uid >= 1000 quiet
auth                required        pam_deny.so
...

```

You should be able to login as root after restarting GDM.

Hide user from login list

The users for the gdm user list are gathered by [AccountsService](#). It will automatically hide system users (UID < 1000). To hide ordinary users from the login list create or edit a file named after the user to hide in `/var/lib/AccountsService/users/` to contain at least:

```

/var/lib/AccountsService/users/<username>

[User]
SystemAccount=true

```

Setup default monitor settings

Some [desktop environments](#) store display settings in `~/.config/monitors.xml`. `xrandr` commands are then generated on the base of the file content. GDM has a similar file stored in `/var/lib/gdm/.config/monitors.xml`.

If you have your monitors setup as you like (orientation, primary and so on) in `~/.config/monitors.xml` and want GDM to honor those settings:

```
# cp ~/.config/monitors.xml /var/lib/gdm/.config/monitors.xml
```

Changes will take effect on logout. This is necessary because GDM does not respect `xorg.conf`.

Note: If you use GDM under Wayland, you must also use a `monitors.xml` that was created under Wayland. See [GNOME bug 748098](#) for more info. Alternatively, you can force GDM to [#Use Xorg backend](#), and use a `monitors.xml` that was created under Xorg.

Configure X server access permission

You can use the `xhost` command to configure X server access permissions.

For instance, to grant GDM the right to access the X server, use the following command:

```
# xhost +SI:localuser:gdm
```

Troubleshooting

Failure to use proprietary NVIDIA driver

GDM uses the [Wayland](#) backend by default which conflicts with NVIDIA driver. Turning off the Wayland backend could enable proprietary NVIDIA driver.

Failure on logout

If GDM starts up properly on boot, but fails after repeated attempts on logout, try adding this line to the daemon section of `/etc/gdm/custom.conf`:

```
GdmXserverTimeout=60
```

Rootless Xorg

See [Xorg#Rootless Xorg](#).

Use Xorg backend

The [Wayland](#) backend is used by default and the [Xorg](#) backend is used only if the Wayland backend cannot be started. As the Wayland backend has been [reported](#) to cause problems for some users, use of the Xorg backend may be necessary. To use the Xorg backend by default, edit the `/etc/gdm/custom.conf` file and uncomment the following line:

```
#WaylandEnable=false
```

Incomplete removal of gdm

After removing [gdm](#), [systemd](#) may report the following:

```
user 'gdm': directory '/var/lib/gdm' does not exist
```

To remove this warning, login as root and delete the primary user "gdm" and then delete the group "gdm":

```
# userdel gdm
```

```
# groupdel gdm
```

Verify that gdm is successfully removed via `pwck` and `grpck`. To round it off, you may want to double-check no [unowned files](#) for gdm remain.

See also

- [GDM Reference Manual](#)