Short help/information for Xorg wintersky 2021.3.0

The Xorg and noX flavors come with a new tool to connect your wireless devices. Intels *iNet wireless daemon* (***iwd***) sends WPA-Supplicant into well-earned retirement. Ten times smaller and a lot faster, ***iwd*** will be the successor. Further information can be found in the *Arch Linux wiki*.

You can start an interactive shell by running ***iwctl*** from a terminal. Entering 'help' in the shell gives you all the options to list, scan and connect to your device or simply use ***nmtui*** or ***nmcli*** on the command line or the graphical-interface of Network-Manager

You can also use the following command to connect, without the interaktiv shell, to your Wi-Fi:

```
iwctl --passphrase T0pS3ckr3t station wlan0 connect
Home-Office
```

If it is a hidden network, use the ***'connect-hidden'*** option.
To find out which station you need to point out, use the ***'iwctl station list'*** command.

If you want to use ***wpa_supplicant*** instead of ***iwd*** follow the steps below.

First stop and mask iwd.service, stop NetworkManager.service, rename /etc/NetworkManager/conf.d/nm.conf, unmask and start wpa_supplicant.service, start NetworkManager.service again:

```
sudo systemctl stop iwd.service
sudo systemctl mask iwd.service
sudo systemctl stop NetworkManager.service
sudo mv /etc/NetworkManager/conf.d/nm.conf
/etc/NetworkManager/conf.d/nm.conf~
sudo systemctl unmask wpa_supplicant.service
sudo systemctl enable --now wpa_supplicant.service
sudo systemctl start NetworkManager.service
```

Now you should be able to connect your wireless device with wpa_supplicant.

Installing iwd

If you like, you can use **iwd** in the other flavors too, either on its own or within Network-Manager.

To easily set iwd up, use these steps:

```
sudo apt update
sudo apt install iwd
sudo systemctl stop wpa_supplicant.service
sudo systemctl mask wpa_supplicant.service
sudo systemctl stop NetworkManager.service
sudo touch /etc/NetworkManager/conf.d/nm.conf
sudo echo -e "[device]\nwifi.backend=iwd" >
/etc/NetworkManager/conf.d/nm.conf
sudo touch /etc/iwd/main.conf
sudo echo -e
"[General]\nEnableNetworkConfiguration=true
\n\n[Network]\nNameResolvingService=systemd" >
/etc/iwd/main.conf
sudo systemctl enable --now iwd.service
sudo systemctl start NetworkManager.service
```

The following non-free and contrib packages are installed by default:

Non-free

- amd64-microcode - Processor microcode firmware for AMD CPUs
- firmware-amd-graphics - Binary firmware for AMD/ATI graphics chips
- firmware-atheros - Binary firmware for Atheros wireless cards
- firmware-bnx2 - Binary firmware for Broadcom NetXtremeII
- firmware-bnx2x - Binary firmware for Broadcom NetXtreme II 10Gb
- firmware-brcm80211 - Binary firmware for Broadcom 802.11 wireless card
- firmware-crystalhd - Crystal HD Video Decoder (firmware)
- firmware-intelwimax - Binary firmware for Intel WiMAX Connection
- firmware-iwlwifi - Binary firmware for Intel Wireless cards
- firmware-libertas - Binary firmware for Marvell Libertas 8xxx wireless card
- firmware-linux-nonfree - Binary firmware for various drivers in the Linux kernel
- firmware-misc-nonfree - Binary firmware for various drivers in the Linux kernel
- firmware-myricom - Binary firmware for Myri-10G Ethernet adapters
- firmware-netxen - Binary firmware for QLogic Intelligent Ethernet (3000)
- firmware-qlogic - Binary firmware for QLogic HBAs
- firmware-realtek - Binary firmware for Realtek wired/wifi/BT adapters
- firmware-ti-connectivity - Binary firmware for TI Connectivity wireless network
- firmware-zd1211 - binary firmware for the zd1211rw wireless driver
- intel-microcode - Processor microcode firmware for Intel CPUs

Contrib

- b43-fwcutter - utility for extracting Broadcom 43xx firmware
- firmware-b43-installer - firmware installer for the b43 driver
- firmware-b43legacy-installer - firmware installer for the b43legacy driver
- iucode-tool - Intel processor microcode

Return to non-free

Right now the installer does not offer an opt-out for packages that do not comply with DFSG, the Debian Free Software Guidelines. That means that non-free packages like unfree firmware will be installed by default on the system. The command **vrms** will list these packages for you. You can remove not wanted packages manually or remove them all by issuing **apt purge $(vrms -s)** before or after installation. Else our script **remove-nonfree** can do that for you.

Installation hints and known issues

- If you want to reuse your existing home (or any other data) partition, it would be wise to do so after installation and not in the Calamares Installer.
- Encryption setups like LUKS and others are not yet supported by the Installer Framework, you should prepare your partition setup beforehand and might better use the **cli-installer** in a terminal.
- With some Intel GPUs on some devices, you might encounter a frozen system on Live shortly after boot. To fix this you need to set the kernel parameter **intel_iommu=igfx_off** before booting again.

Credits for siduction 2021.3.0

Core Team

Alf Gaida (agaida), vanished from the face of the earth
Axel Beu (ab) †, In memoriam
Torsten Wohlfarth (towo)
Hendrik Lehmbruch (hendrikL)
Ferdinand Thommes (devil)
Vinzenz Vietzke (vinzv)

Code, ideas and support:

der_bud
Stefan Tell (cryptosteve)
Markus Meyer (coruja)
akli (for his work on getting the manual back in shape)

Thank you!

We want to thank all testers and all the people giving us support over the years. This is also your achievement. We also want to thank Debian, as we are standing on the shoulders of giants.
And now enjoy!

On behalf of the siduction team:

Ferdinand Thommes