

# Configuring the network

From Gentoo Wiki

< [Handbook:AMD64 \(/wiki/Special:MyLanguage/Handbook:AMD64\)](#) | [Installation \(/wiki/Special:MyLanguage/Handbook:AMD64/Installation\)](#)

[Jump to:navigation](#) [Jump to:search](#)

## Contents

- 1 Automatic network configuration
  - 1.1 Using DHCP
  - 1.2 Testing the network
- 2 Obtaining interface info
- 3 Optional: Application specific configuration
  - 3.1 Configure web proxies
  - 3.2 Using pppoe-setup for ADSL
  - 3.3 Using PPTP
  - 3.4 Configuring WEP
- 4 Using net-setup
- 5 Internet and IP basics
  - 5.1 Interfaces and addresses
  - 5.2 Networks and CIDR
  - 5.3 The Internet
  - 5.4 The Domain Name System
- 6 Manual network configuration
  - 6.1 Interface address configuration
  - 6.2 Default route configuration
  - 6.3 DNS configuration

## Automatic network configuration


Maybe it just works?

If the system is connected to an Ethernet network with an IPv6 router or DHCP server, it's very likely that the system's network was configured automatically. If additional, advanced configuration is not required, Internet connectivity can be tested ([/wiki/Handbook:AMD64/Installation/Networking#Testing\\_the\\_network](#)).

## Using DHCP

DHCP (Dynamic Host Configuration Protocol) assists in network configuration, and can automatically provide configuration for a variety of parameters including: IP address, network mask, routes, DNS servers, NTP servers, etc.

DHCP requires that a server be running on the same *Layer 2 (Ethernet)* segment as the client requesting a *lease*. DHCP is often used on RFC1918 (*private*) networks, but is also used to acquire public IP information from ISPs.

 **Tip**

Official Gentoo *boot media* runs **dhcpcd** automatically at startup. This behavior can be disabled by adding the `nodhcp` argument to the *boot media* kernel commandline ([/wiki/Handbook:AMD64/Installation/Media](#)).

If it is not already running, dhcpcd can be started on **enp1s0** with:

```
root # dhcpcd enp1s0
```

Some network administrators require that the hostname and domain name provided by the DHCP server is used by the system. In that case, use:

```
root # dhcpcd -HD enp1s0
```

**AMD64 Handbook (/wiki/Handbook:AMD64)**  
**Installation (/wiki/Handbook:AMD64/Full/Installation)**  
About the installation (/wiki/Handbook:AMD64/Installation/About)  
Choosing the media (/wiki/Handbook:AMD64/Installation/Media)  
Configuring the network  
Preparing the disks (/wiki/Handbook:AMD64/Installation/Disks)  
The stage file (/wiki/Handbook:AMD64/Installation/Stage)  
Installing base system (/wiki/Handbook:AMD64/Installation/Base)  
Configuring the kernel (/wiki/Handbook:AMD64/Installation/Kernel)  
Configuring the system (/wiki/Handbook:AMD64/Installation/System)  
Installing tools (/wiki/Handbook:AMD64/Installation/Tools)  
Configuring the bootloader (/wiki/Handbook:AMD64/Installation/Bootloader)  
Finalizing (/wiki/Handbook:AMD64/Installation/Finalizing)  
**Working with Gentoo (/wiki/Handbook:AMD64/Full/Working)**  
Portage introduction (/wiki/Handbook:AMD64/Working/Portage)  
USE flags (/wiki/Handbook:AMD64/Working/USE)  
Portage features (/wiki/Handbook:AMD64/Working/Features)  
Initscript system (/wiki/Handbook:AMD64/Working/Initscripts)  
Environment variables (/wiki/Handbook:AMD64/Working/EnvVar)  
**Working with Portage (/wiki/Handbook:AMD64/Full/Portage)**

To stop **dhcpcd**, **-x** can be used:

```
root # dhcpcd -x

sending signal Term to pid 10831
waiting for pid 10831 to exit
```

**See also**  
Dhcpcd usage (/wiki/Dhcpcd#Usage)

## Testing the network

A properly configured *default* route is a critical component of Internet connectivity, route configuration can be checked with:

```
root # ip route

default via 192.168.0.1 dev enp1s0
```

If no *default* route is defined, Internet connectivity is unavailable, and additional configuration is required.

Basic internet connectivity can be confirmed with a ping:

```
root # ping -c 3 1.1.1.1
```

**Tip**  
It's helpful to start by pinging a known IP address instead of a hostname. This can isolate DNS issues from basic Internet connectivity issues.

Outbound HTTPS access and DNS resolution can be confirmed with:

```
root # curl --location gentoo.org --output /dev/null

Unless curl reports an error, or other tests fail, the installation process can be continued with disk preparation (/wiki/Handbook:AMD64/Installation/Disks).
```

If **curl** reports an error, but Internet-bound pings work, DNS may need configuration (/wiki/Handbook:AMD64/Installation/Networking#DNS\_configuration).

If Internet connectivity has not been established, first interface information should be verified (/wiki/Handbook:AMD64/Installation/Networking#Obtaining\_interface\_info), then:

- net-setup can be used (/wiki/Handbook:AMD64/Installation/Networking#Using\_net-setup) to assist in network configuration.
- Application specific configuration (/wiki/Handbook:AMD64/Installation/Networking#Application\_specific\_configuration) may be required.
- Manual network configuration (/wiki/Handbook:AMD64/Installation/Networking#Manual\_network\_configuration) can be attempted.

## Obtaining interface info

If networking doesn't work out of the box, additional steps must be taken to enable Internet connectivity. Generally, the first step is to enumerate host network interfaces.

The **ip** command, part of the sys-apps/iproute2 (https://packages.gentoo.org/packages/sys-apps/iproute2) package, can be used to query and configure system networking.

The **link** argument can be used to display network interface links:

```
root # ip link

1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default qlen 1000
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
4: enp1s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP mode DEFAULT group default qlen 1000
   link/ether e8:40:f2:ac:25:7a brd ff:ff:ff:ff:ff:ff
```

The **address** argument can be used to query device address information:

```
root # ip address
```

Files and directories (/wiki/Handbook:AMD64/Portage/Files)
Variables (/wiki/Handbook:AMD64/Portage/Variables)
Mixing software branches (/wiki/Handbook:AMD64/Portage/Branches)
Additional tools (/wiki/Handbook:AMD64/Portage/Tools)
Custom package repository (/wiki/Handbook:AMD64/Portage/CustomTree)
Advanced features (/wiki/Handbook:AMD64/Portage/Advanced)
<b>OpenRC network configuration (/wiki/Handbook:AMD64/Full/Networking)</b>
Getting started (/wiki/Handbook:AMD64/Networking/Introduction)
Advanced configuration (/wiki/Handbook:AMD64/Networking/Advanced)
Modular networking (/wiki/Handbook:AMD64/Networking/Modular)
Wireless (/wiki/Handbook:AMD64/Networking/Wireless)
Adding functionality (/wiki/Handbook:AMD64/Networking/Extending)
Dynamic management (/wiki/Handbook:AMD64/Networking/Dynamic)

```
2: enp1s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
<pre>
    link/ether e8:40:f2:ac:25:7a brd ff:ff:ff:ff:ff:ff
    inet 10.0.20.77/22 brd 10.0.23.255 scope global enp1s0
        valid_lft forever preferred_lft forever
    inet6 fe80::ea40:f2ff:feac:257a/64 scope link
        valid_lft forever preferred_lft forever
```

The output of this command contains information for each network interface on the system. Entries begin with the device index, followed by the device name: **enp1s0**.

#### Tip

If no interfaces other than the **lo** (*loopback*) are displayed, then the networking hardware is faulty, or the driver for the interface has not been loaded into the kernel. Both situations reach beyond the scope of this Handbook. Please ask for support in contact **#gentoo** ([ircs://irc.libera.chat/#gentoo](irc://irc.libera.chat/#gentoo)) (webchat (<https://web.libera.chat/#gentoo>)).

For consistency, the handbook will assume that the primary network interface is called **enp1s0**.

#### Note

As a result of the shift toward predictable network interface names (<https://www.freedesktop.org/wiki/Software/systemd/PredictableNetworkInterfaceNames/>), the interface name on the system can be quite different than the old **eth0** naming convention. Modern Gentoo boot media uses interface names with prefixes such as **eno0**, **ens1**, or **enp5s0**.

## Optional: Application specific configuration

The following methods are not generally required, but may be helpful in situations where additional configuration is required for Internet connectivity.

### Configure web proxies

If the internet is accessed through a web proxy, then it will be necessary to define proxy information to for Portage to properly access the proxy for each supported protocol. Portage observes the *http\_proxy*, *ftp\_proxy*, and *RSYNC\_PROXY* environment variables in order to download packages via its **wget** and **rsync** retrieval mechanisms.

Certain text-mode web browsers such as **links** can also make use of environment variables that define web proxy settings; in particular for the HTTPS access it also will require the *https\_proxy* environment variable to be defined. While Portage will be influenced without passing extra run time parameters during invocation, **links** will require proxy settings to be set.

In most cases, it is sufficient to define environment variables using the server hostname. In the following example, it is assumed the proxy server host is called *proxy.gentoo.org* and the port is 8080.

#### Note

The **#** symbol in the following commands is a comment. It has been added for clarity only and does *not* need to be typed when entering the commands.

To define an HTTP proxy (for HTTP and HTTPS traffic):

```
root # export http_proxy="http://proxy.gentoo.org:8080" # Applies to Portage and Links
root # export https_proxy="http://proxy.gentoo.org:8080" # Only applies for Links
```

If the HTTP proxy requires authentication, set a username and password with the following syntax:

```
root # export http_proxy="http://username:password@proxy.gentoo.org:8080" # Applies to Portage and Links
root # export https_proxy="http://username:password@proxy.gentoo.org:8080" # Only applies for Links
```

Start **links** using the following parameters for proxy support:

```
user $ links -http-proxy ${http_proxy} -https-proxy ${https_proxy}
```

To define an FTP proxy for Portage and/or **links**:

```
root # export ftp_proxy="ftp://proxy.gentoo.org:8080" # Applies to Portage and Links
```

Start **links** using the following parameter for a FTP proxy:

```
user $ links -ftp-proxy ${ftp_proxy}
```

To define an RSYNC proxy for Portage:

```
root # export RSYNC_PROXY="proxy.gentoo.org:8080" # Applies to Portage; Links does not support a rsync proxy
```

### Using pppoe-setup for ADSL

If PPPoE is required for Internet access, the Gentoo *boot media* includes the **pppoe-setup** script to simplify **ppp** configuration.

During setup, **pppoe-setup** will ask for:

- The name of the Ethernet *interface* connected to the ADSL modem.
- The PPPoE username and password.
- DNS server IPs.
- Whether or not a firewall is needed.

```
root # pppoe-setup
```

```
root # pppoe-start
```

In the event of failure, credentials in `/etc/ppp/pap-secrets` or `/etc/ppp/chap-secrets` should be verified. If credentials are correct, PPPoE Ethernet interface selection should be checked.

## Using PPTP

If PPTP support is needed, **pptpclient** can be used, but requires configuration prior to usage.

Edit `/etc/ppp/pap-secrets` or `/etc/ppp/chap-secrets` so it contains the correct username/password combination:

```
root # nano /etc/ppp/chap-secrets
```

Then adjust `/etc/ppp/options.pptp` if necessary:

```
root # nano /etc/ppp/options.pptp
```

Once configuration is complete, run **pptp** (along with the options that couldn't be set in `options.pptp`) to connect the server:

```
root # pptp <server ipv4 address>
```

## Configuring WEP

### Warning

Do not use WEP unless it is the only option. WEP provides essentially no security over an open network.

### Important

The **iw** command is only available on the following architectures: **amd64**, **x86**, **arm**, **arm64**, **ppc**, **ppc64**, and **riscv**.

When using a wireless (802.11) card, the wireless settings need to be configured before going any further. To see the current wireless settings on the card, one can use **iw**. Running **iw** might show something like:

```
root # iw dev wlp9s0 info
```

```
Interface wlp9s0
    ifindex 3
    wdev 0x1
    addr 00:00:00:00:00:00
    type managed
    wiphy 0
    channel 11 (2462 MHz), width: 20 MHz (no HT), center1: 2462 MHz
    txpower 30.00 dBm
```

To check for a current connection:

```
root # iw dev wlp9s0 link
```

```
Not connected.
```

or

```
root # iw dev wlp9s0 link
```

```
Connected to 00:00:00:00:00:00 (on wlp9s0)
    SSID: GentooNode
    freq: 2462
    RX: 3279 bytes (25 packets)
    TX: 1049 bytes (7 packets)
    signal: -23 dBm
    tx bitrate: 1.0 MBit/s
```

### Note

Some wireless cards may have a device name of **wlan0** or **ra0** instead of **wlp9s0**. Run **ip link** to determine the correct device name.

For most users, there are only two settings needed to connect, the ESSID (aka wireless network name) and, optionally, the WEP key.

- First, ensure the interface is active:

```
root # ip link set dev wlp9s0 up
```

- To connect to an open network with the name *GentooNode*:

```
root # iw dev wlp9s0 connect -w GentooNode
```

- To connect with a hex WEP key, prefix the key with `d:`:

```
root # iw dev wlp9s0 connect -w GentooNode key 0:d:1234123412341234abcd
```

- To connect with an ASCII WEP key:

```
root # iw dev wlp9s0 connect -w GentooNode key 0:some-password
```

#### Note

If the wireless network is set up with WPA or WPA2, then **wpa\_supplicant** needs to be used. For more information on configuring wireless networking in Gentoo Linux, please read the Wireless networking chapter (</wiki/Handbook:AMD64/Networking/Wireless>) in the Gentoo Handbook.

Confirm the wireless settings by using **iw dev wlp9s0 link**. Once wireless is working, continue configuring the IP level networking options as described in the next section (Understanding network terminology ([/wiki/Handbook:AMD64/Installation/Networking#Understanding\\_network\\_terminology](/wiki/Handbook:AMD64/Installation/Networking#Understanding_network_terminology))) or use the **net-setup** tool as described previously.

## Using net-setup

In cases where automatic network configuration is unsuccessful, the Gentoo *boot media* provides scripts to aid in network configuration. **net-setup** can be used to configure wireless network information and static IPs.

```
root # net-setup enp1s0
```

**net-setup** will ask some questions about the network environment and will use that information to configure **wpa\_supplicant** or *static addressing*.

#### Important

Network status should be tested ([/wiki/Handbook:AMD64/Installation/Networking#Testing\\_the\\_network](/wiki/Handbook:AMD64/Installation/Networking#Testing_the_network)) after any configuration steps are taken. In the event that configuration scripts do not work, manual network configuration ([/wiki/Handbook:AMD64/Installation/Networking#Manual\\_network\\_configuration](/wiki/Handbook:AMD64/Installation/Networking#Manual_network_configuration)) is required.

## Internet and IP basics

If all of the above fails, the network must be configured manually. This is not particularly difficult, but should be done with consideration. This section serves to clarify terminology and introduce users to basic networking concepts pertaining to manually configuring an Internet connection.

#### Tip

Some **CPE (Carrier Provided Equipment)** combines the functions of a *router*, *access point*, *modem*, *DHCP server*, and *DNS server* into one unit. It's important to differentiate the functions of a device from the physical appliance.

## Interfaces and addresses

Network *interfaces* are logical representations of network devices. An *interface* needs an *address* to communicate with other devices on the *network*. While only a single *address* is required, multiple addresses can be assigned to a single *interface*. This is especially useful for dual stack (IPv4 + IPv6) configurations.

For consistency, this primer will assume the interface **enp1s0** will be using the address 192.168.0.2.

#### Important

IP addresses can be set arbitrarily. As a result, it's possible for multiple devices to use the same IP address, resulting in an *address conflict*. Address conflicts should be avoided by using DHCP or SLAAC.

#### Tip

IPv6 typically uses **StateLess Address AutoConfiguration (SLAAC)** for address configuration. In most cases, manually setting IPv6 addresses is a bad practice. If a specific address suffix is preferred, interface identification tokens ([/wiki/IPv6\\_Static\\_Addresses\\_using\\_Tokens](/wiki/IPv6_Static_Addresses_using_Tokens)) can be used.

## Networks and CIDR

Once an address is chosen, how does a device know how to talk to other devices?

IP *addresses* are associated with *networks*. IP *networks* are contiguous logical ranges of addresses.

*Classless Inter-Domain Routing* or *CIDR* notation is used to distinguish network sizes.

- The *CIDR* value, often notated starting with a */*, represents the size of the network.
  - The formula  $2^{(32 - CIDR)}$  can be used to calculate network size.
  - Once network size is calculated, usable node count must be reduced by 2.
    - The first IP in a network is the *Network address*, and the last is typically the *Broadcast address*. These addresses are special and cannot be used by normal hosts.



### Tip

The most common *CIDR* values are **/24**, and **/32**, representing **254** nodes and a single node respectively.

A *CIDR* of **/24** is the de-facto default network size. This corresponds to a subnet mask of 255.255.255.0, where the last 8 bits are reserved for IP addresses for nodes on a network.

The notation: **192.168.0.2/24** can be interpreted as:

- The *address* **192.168.0.2**
- On the *network* 192.168.0.0
- With a size of **254** ( $2^{(32 - 24)} - 2$ )
  - Usable IPs are in the range 192.168.0.1 - 192.168.0.254
- With a *broadcast address* of 192.168.0.255
  - In most cases, the last address on a network is used as the *broadcast address*, but this can be changed.

Using this configuration, a device should be able to communicate with any host on the same *network* (**192.168.0.0**).

## The Internet

Once a device is on a network, how does it know how to talk to devices on the Internet?

To communicate with devices outside of local *networks*, *routing* must be used. A *router* is simply a network device that forwards traffic for other devices. The term *default route* or *gateway* typically refers to whatever device on the current network is used for external network access.



### Tip

It's a standard practice to make the *gateway* the first or last IP on a network.

If an Internet-connected router is available at **192.168.0.1**, it can be used as the *default route*, granting Internet access.

To summarize:

- Interfaces must be configured with an *address* and *network information*, such as the *CIDR* value.
- Local network access is used to access a *router* on the same network.
- The *default route* is configured, so traffic destined for external networks is forwarded to the *gateway*, providing Internet access.

## The Domain Name System

Remembering IPs is hard. The *Domain Name System* was created to allow mapping between *Domain Names* and *IP addresses*.

Linux systems use **/etc/resolv.conf** to define *nameservers* to be used for *DNS resolution*.



### Tip

Many *routers* can also function as a DNS server, and using a local DNS server can augment privacy and speed up queries through caching.

Many ISPs run a DNS server that is generally advertised to the *gateway* over DHCP. Using a local DNS server tends to improve query latency, but most public DNS servers will return the same results, so server usage is largely based on preference.

## Manual network configuration

### Interface address configuration



### Important

When manually configuring IP addresses, the local network topology must be considered. IP addresses can be set arbitrarily; conflicts may cause network disruption.

To configure **enp1s0** with the address **192.168.0.2** and CIDR /24:

```
root # ip address add 192.168.0.2/24 dev enp1s0
```

**Tip**

The start of this command can be shortened to **ip a**.

## Default route configuration

Configuring address and network information for an interface will configure **link** routes, allowing communication with that network segment:

```
root # ip route
```

```
192.168.0.0/24 dev enp1s0 proto kernel scope link src 192.168.0.2
```

**Tip**

This command can be shortened to **ip r**.

The **default** route can be set to **192.168.0.1** with:

```
root # ip route add default via 192.168.0.1
```

## DNS configuration

Nameserver info is typically acquired using DHCP, but can be set manually by adding `nameserver` entries to **/etc/resolv.conf**.

**Warning**

If `dhcpcd` is running, changes to **/etc/resolv.conf** will not persist. Status can be checked with `ps x | grep dhcpcd`.

**nano** is included in Gentoo *boot media* and can be used to edit **/etc/resolv.conf** with:

```
root # nano /etc/resolv.conf
```

Lines containing the keyword `nameserver` followed by a DNS server IP address are queried in order of definition:

**FILE** **/etc/resolv.conf** **Use Quad9 DNS.**

```
nameserver 9.9.9.9
nameserver 149.112.112.112
```

**FILE** **/etc/resolv.conf** **Use Cloudflare DNS.**

```
nameserver 1.1.1.1
nameserver 1.0.0.1
```

DNS status can be checked by pinging a domain name:

```
root # ping -c 3 gentoo.org
```

Once connectivity has been verified ([/wiki/Handbook:AMD64/Installation/Networking#Testing\\_the\\_network](/wiki/Handbook:AMD64/Installation/Networking#Testing_the_network)), continue with Preparing the disks (</wiki/Handbook:AMD64/Installation/Disks>).

← [Choosing the media \(/wiki/Handbook:AMD64/Installation/Media\)](/wiki/Handbook:AMD64/Installation/Media) [Installation/Networking](/wiki/Handbook:AMD64/Installation/Networking) → [Preparing the disks \(/wiki/Handbook:AMD64/Installation/Disks\)](/wiki/Handbook:AMD64/Installation/Disks)

Retrieved from "<https://wiki.gentoo.org/index.php?title=Handbook:AMD64/Installation/Networking&oldid=212412> (<https://wiki.gentoo.org/index.php?title=Handbook:AMD64/Installation/Networking&oldid=212412>)"

- This page was last edited on 2 January 2015, at 00:03.
- [Privacy policy \(/wiki/Gentoo\\_Wiki:Privacy\\_policy\)](/wiki/Gentoo_Wiki:Privacy_policy)
- [About Gentoo Wiki \(/wiki/Gentoo\\_Wiki:About\)](/wiki/Gentoo_Wiki:About)
- [Disclaimers \(/wiki/Gentoo\\_Wiki:General\\_disclaimer\)](/wiki/Gentoo_Wiki:General_disclaimer)

© 2001–2024 Gentoo Authors

Gentoo is a trademark of the Gentoo Foundation, Inc. and of Förderverein Gentoo e.V. The contents of this document, unless otherwise expressly stated, are licensed under the CC-BY-SA-4.0 (<https://creativecommons.org/licenses/by-sa/4.0/>) license. The [Gentoo Name and Logo Usage Guidelines \(https://www.gentoo.org/inside-gentoo/foundation/name-logo-](https://www.gentoo.org/inside-gentoo/foundation/name-logo-)

[guidelines.html](#)) apply.

---