**How To Create Static Routes to Route IP Traffic**

By default all network traffic will normally be configured to route via the default gateway, that is the router attached to the network interface. It may not always be the case that you want all traffic to take the same path, in these instances we can set additional static routes that will forward specific traffic out of a different interface rather than the default gateway. This may be required if you need to be able to reach a particular network that your default gateway router does not know about.

Here we’re going to cover how to configure a static route in Linux.  
  
Please be advised that this example is done within CentOS 7 so some information may vary slightly depending on your distribution of Linux.

**Viewing the Routing Table**

There are a few ways that you can view the routing table, with either the older ‘route’ command, or the newer ‘ip route’ command as shown below.

[root@centos7 ~]# **route**

Kernel IP routing table

Destination Gateway Genmask Flags Metric Ref Use Iface

default 192.168.1.254 0.0.0.0 UG 100 0 0 eno16777736

192.168.1.0 0.0.0.0 255.255.255.0 U 100 0 0 eno16777736

[root@centos7 ~]# **ip route**

default via 192.168.1.254 dev eno16777736 proto static metric 100

192.168.1.0/24 dev eno16777736 proto kernel scope link src 192.168.1.10 metric 100

Basically this output is showing that the default route here is 192.168.1.254 and all traffic (0.0.0.0/0) is sent here, this is standard and default configuration. If we want to send traffic to a particular network range elsewhere a static route can be used. Traffic will be sent to the route in the routing table that is most specific to it.

**Setting a Static Route**

To set a static route we need to know the some information about the network that we are trying to route to, such as the IP address range and netmask, the IP address of the gateway, and possibly the metric. In our example here we will be creating a static route to send traffic destined to the 10.0.0.0/8 network to 10.0.0.1

In our particular network configuration let’s say that the router at 192.168.1.254 does not know anything about the 10.0.0.0/8 network at all and is not able to communicate with it. When our CentOS 7 server at 192.168.1.10 tries to reach something in the 10.0.0.0/8 network the routing table will forward the traffic to the router at the default gateway of 192.168.1.254, however in our example this router does not know where to forward this traffic next, depending on the configuration it may simply attempt to use its default gateway if there are no other routes available or drop it.

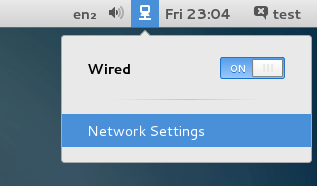
To prevent this behaviour we can configure a static route on the CentOS 7 server which will allow us to manually specify where we want to send traffic destined to the 10.0.0.0/8 network.

There are a few different ways a static route can be set, we’ll cover a few options below. These all achieve the same result so you can use which ever option is easiest for you.

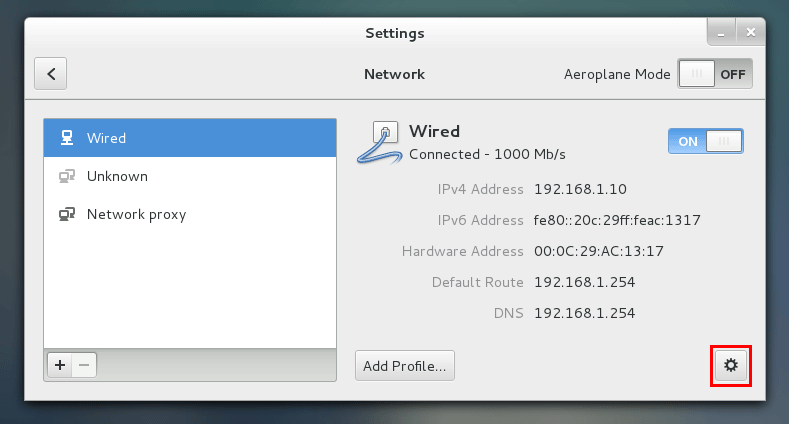
**Using the Network Manager GUI**

If you have a graphical user interface (GUI) installed this option is probably the easiest for you to complete.

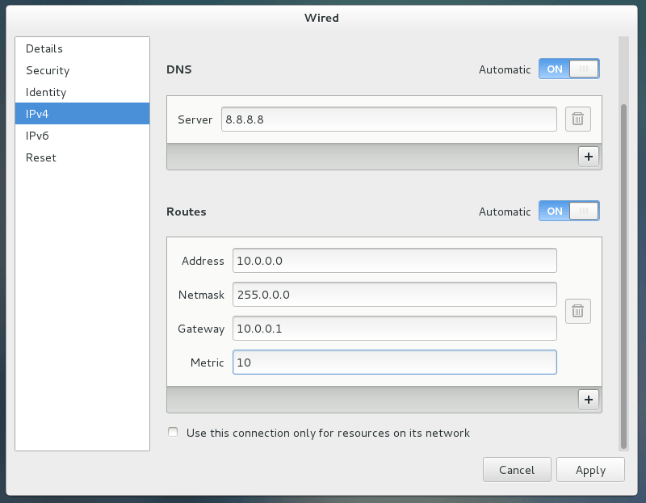
First open the Network Settings, this can be found by clicking the network icon up the top right.

[](https://www.rootusers.com/wp-content/uploads/2015/09/open-network-settings.png)

Next select the particular network connection that the static route is for, in this case we’re going to edit the Wired network. After selecting this, click the cog icon as highlighted below to open the settings.

[](https://www.rootusers.com/wp-content/uploads/2015/09/network-settings.png)

Next select either IPv4 or IPv6 from the menu on the left, then scroll down to the Routes section and enter the routing information and click Apply.

[](https://www.rootusers.com/wp-content/uploads/2015/09/static-route.png)

Once you click the Apply button a file will be created within /etc/sysconfig/network-scripts/route-<InterfaceName>, for example after specifying the configuration in the images above the following configuration file was created.

[root@centos7 ~]# **cat /etc/sysconfig/network-scripts/route-eno16777736**

ADDRESS0=10.0.0.0

NETMASK0=255.0.0.0

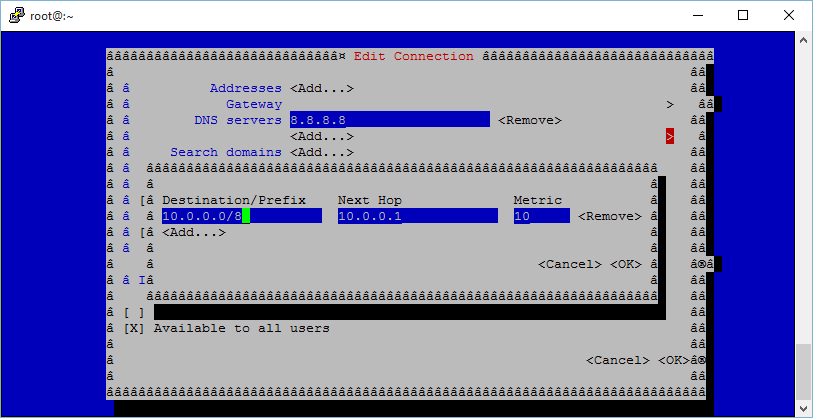
GATEWAY0=10.0.0.1

METRIC0=10

In this case our interface name is eno16777736

**Using the Network Manager TUI**

This provides you with the same options as the GUI option above and will create the static configuration file afterwards, the difference is that the configuration is done through a text user interface (TUI) meaning that you can do it through a terminal as pictured below.

[](https://www.rootusers.com/wp-content/uploads/2015/09/network-settings-tui.png)

**Using the nmcli Command**

The nmcli command can also be used to manage the network completely by command line, a quick example is shown below of setting a static route with this tool.

[root@centos7 ~]# **nmcli con edit type ethernet con-name eno16777736**

===| nmcli interactive connection editor |===

Adding a new '802-3-ethernet' connection

Type 'help' or '?' for available commands.

Type 'describe [.]' for detailed property description.

You may edit the following settings: connection, 802-3-ethernet (ethernet), 802-1x, ipv4, ipv6, dcb

nmcli> **set ipv4.routes 10.0.0.0/8 10.0.0.1**

nmcli> **save persistent**

Saving the connection with 'autoconnect=yes'. That might result in an immediate activation of the connection.

Do you still want to save? (yes/no) [yes] yes

Connection 'eno16777736' (e4f5af01-1c77-4240-ac9c-1bfa4b8a1d3c) successfully saved.

nmcli> **quit**

The IP command can also be used to configure static routes through command line, however these changes are not persistently stored so it is recommended to use nmcli instead with ‘save persistent’.

**Manually Edit the Configuration File**

As we have seen the previous options simply end up modifying files on disk, so it is also possible to modify these manually with a text editor. Just be careful however as a connection that is managed by network manager may overwrite manual changes that you have made.

**Applying the Changes**

Note that after making changes to the network configuration you will need to restart the network to apply the changes, this can be done in the GUI by clicking the on/off button on the interface or otherwise with the following commands.

systemctl restart network

We can also bring the interface down and back up with nmcli or ifdown/ifup, replacing eno16777736 with the name of your interface. If you are unsure of your interface name you can use the ‘ip addr’ command.

nmcli con down eno16777736; nmcli con up eno16777736

OR

ifdown eno16777736; ifup eno16777736

Note the ; after the first command, this will basically run the nmcli con up or ifup command straight after the interface is brought down. While this doesn’t matter too much if you’re working through the console, if you are doing this over SSH for instance if you take the interface down you will be disconnected and not be able to bring it back up. Running both commands at once in this manner ensures that the interface will come back up, assuming of course there are no problems with your network configuration changes.

**Summary**

As shown there are many different ways that allow you to define a static route in Linux which is useful if you need to control where specific network traffic should be routed to.