

# Adding Multiple IP Addresses in CentOS 6



**Note:** In this article we are using private IP addresses (192.168.0.0) as an example. On your servers you'll more than likely be adding IP addresses that are publicly routed and available. If you wish to purchase additional IPs for your 100TB bare metal server, feel free to contact our Sales team, they'll be happy to help you out.

## Introduction

The task of assigning multiple IP addresses to a single NIC is a fairly common one. Fortunately, it's a rather easy process and shouldn't take more than a few minutes. This how-to guide will show you the process behind adding additional IPs to your NIC.

## View currently assigned IPs

Before we begin, we'll need to see the IP(s) that are currently assigned to the server. To do so, run the following command:

```
ifconfig -a
```

The output should appear similar to the following:

```
p2p1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.0.20 netmask 255.255.255.0 broadcast 192.168.0.255
    inet6 fe80::7a45:c4ff:fe17:fa74 prefixlen 64 scopeid 0x20<link>
    ether 78:45:c4:17:fa:74 txqueuelen 1000 (Ethernet)
    RX packets 1689606 bytes 269082641 (256.6 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1795206 bytes 361305113 (344.5 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

**Note:** Your interface might have a different name, such as `eth0`.

You can see the currently assigned IP address by looking for the line that says "inet", and the primary IP address follows. In this case, it's 192.168.0.20. If you're unsure what netmask, broadcast, gateway, etc. are, don't worry, it's not required for the purpose of this guide.

## Creating virtual interfaces (or aliases) and adding additional IPs

Now that we've seen the currently assigned IP and know which interface is configured, we can proceed with assigning additional IPs. This is done through the use of virtual interfaces, or aliases.

You create aliases by setting up additional configuration files in the correct directory. In this case the configuration files for your network interfaces should be found in `/etc/sysconfig/network-scripts/`. You'll need to move to that directory, then run `ls -l`. You should see a list of several files. All of the interface config files start with the prefix "ifcfg-", therefore you should see a file such as "ifcfg-eth0".

Now, when we create an alias, we want to make sure that the primary interface keeps the same settings. To create an additional alias, run the following command. (On my system I used `p2p1`. You need to make sure to use the correct name for your interface such as `ifcfg-eth0`).

```
cp ifcfg-p2p1 ifcfg-p2p1:0
```

Now, if you run `ls -l` again, you should see a new file called "ifcfg-[ifname]:0" ([ifname] should be the name of your interface). Next, you need to open up the new config file in "vi" or another editor such as "nano". You should see several lines like the following:

```
TYPE="Ethernet"
```

```
BOOTPROTO="static"
IPADDR=192.168.0.20
NAME="p2p1"
UUID="113b3841-d06d-4a3c-89e8-1a49105a904a"
ONBOOT=yes
HWADDR="78:45:C4:17:FA:74"
```

The two lines that we want to focus on are highlighted. They are "NAME" and "IPADDR". As I'm sure you've worked out, NAME is the name of the interface, and IPADDR is the IP address currently assigned to that interface. You'll need to change them to the following. Make sure to use your actual IP, not the one I used, as well as the name of your interface.

```
IPADDR=192.168.0.21
NAME="p2p1:0"
```

Once those changes are made, save the file. In order for the changes to take effect, you will need to restart the network service. Run the following command:

```
/etc/init.d/network restart
```

To make sure that this newly created alias is up and running, run the "`ifconfig -a`" command again. You should now see your new alias being shown as "eth0:0" (or whatever your interface name is). To test that your settings are working properly, try to ping the new IP address from outside, using something such as a different server or Windows command prompt.

Congratulations! You've successfully assigned an additional IP address to your network interface!