

11.2. Interface Configuration Files

Interface configuration files control the software interfaces for individual network devices. As the system boots, it uses these files to determine what interfaces to bring up and how to configure them. These files are usually named `ifcfg-name`, where *name* refers to the name of the device that the configuration file controls.

11.2.1. Ethernet Interfaces

One of the most common interface files is `/etc/sysconfig/network-scripts/ifcfg-eth0`, which controls the first Ethernet *network interface card* or NIC in the system. In a system with multiple NICs, there are multiple `ifcfg-ethX` files (where *X* is a unique number corresponding to a specific interface). Because each device has its own configuration file, an administrator can control how each interface functions individually.

The following is a sample `ifcfg-eth0` file for a system using a fixed IP address:

```
DEVICE=eth0 BOOTPROTO=none ONBOOT=yes NETMASK=255.255.255.0 IPADDR=10.0.1.27 USERCTL=no
```

The values required in an interface configuration file can change based on other values. For example, the `ifcfg-eth0` file for an interface using `DHCP` looks different because IP information is provided by the `DHCP` server:

```
DEVICE=eth0 BOOTPROTO=dhcp ONBOOT=yes
```

NetworkManager is graphical configuration tool which provides an easy way to make changes to the various network interface configuration files (see [Chapter 10, NetworkManager](#) (ch-[NetworkManager.html](#)) for detailed instructions on using this tool).

However, it is also possible to manually edit the configuration files for a given network interface.

Below is a listing of the configurable parameters in an Ethernet interface configuration file:

BONDING_OPTS=parameters

sets the configuration parameters for the bonding device, and is used in `/etc/sysconfig/network-scripts/ifcfg-bondN` (see [Section 11.2.4, “Channel Bonding Interfaces”](#) (s2-networkscripts-interfaces-chan.html)). These parameters are identical to those used for bonding devices in `/sys/class/net/bonding_device/bonding`, and the module parameters for the bonding driver as described in *bonding Module Directives*.

This configuration method is used so that multiple bonding devices can have different configurations. In Red Hat Enterprise Linux 6, place all interface-specific bonding options after the `BONDING_OPTS` directive in `ifcfg-name` files. See [Where to specify bonding module parameters](#) (s2-networkscripts-interfaces-chan.html#important-Where_to_specify_bonding_module_parameters) for more information.

BOOTPROTO=protocol

where *protocol* is one of the following:

- › **none** — No boot-time protocol should be used.
- › **bootp** — The `BOOTP` protocol should be used.
- › **dhcp** — The `DHCP` protocol should be used.

BROADCAST=*address*

where *address* is the broadcast address. This directive is deprecated, as the value is calculated automatically with `ipcalc`.

DEVICE=*name*

where *name* is the name of the physical device (except for dynamically-allocated **PPP** devices where it is the *logical name*).

DHCP_HOSTNAME=*name*

where *name* is a short host name to be sent to the **DHCP** server. Use this option only if the **DHCP** server requires the client to specify a host name before receiving an **IP** address.

DHCPV6C=*answer*

where *answer* is one of the following:

- › **yes** — Use **DHCP** to obtain an **IPv6** address for this interface.
- › **no** — Do not use **DHCP** to obtain an **IPv6** address for this interface. This is the default value.

An **IPv6** link-local address will still be assigned by default. The link-local address is based on the MAC address of the interface as per *RFC 4862*.

DHCPV6C_OPTIONS=*answer*

where *answer* is one of the following:

- › **-P** — Enable **IPv6** prefix delegation.
- › **-S** — Use **DHCP** to obtain stateless configuration only, not addresses, for this interface.
- › **-N** — Restore normal operation after using the **-T** or **-P** options.
- › **-T** — Use **DHCP** to obtain a temporary **IPv6** address for this interface.
- › **-D** — Override the default when selecting the type of *DHCP Unique Identifier* (DUID) to use.

By default, the **DHCP** client (dhclient) creates a *DHCP Unique Identifier* (DUID) based on the link-layer address (DUID-LL) if it is running in stateless mode (with the **-s** option, to not request an address), or it creates an identifier based on the link-layer address plus a timestamp (DUID-LLT) if it is running in stateful mode (without **-s**, requesting an address). The **-D** option overrides this default, with a value of either **LL** or **LLT**.

DNS{1,2}=*address*

where *address* is a name server address to be placed in `/etc/resolv.conf` provided that the **PEERDNS** directive is not set to **no**.

ETHTOOL_OPTS=*options*

where *options* are any device-specific options supported by `ethtool`. For example, if you wanted to force 100Mb, full duplex:

```
ETHTOOL_OPTS="autoneg off speed 100 duplex full"
```

Instead of a custom `initscript`, use **ETHTOOL_OPTS** to set the interface speed and duplex settings. Custom `initscripts` run outside of the network init script lead to unpredictable results during a post-boot network service restart.



Set “autoneg off” before changing speed or duplex settings

Changing speed or duplex settings almost always requires disabling auto-negotiation with the `autoneg off` option. This option needs to be stated first, as the option entries are order-dependent.

See [Section 11.8, “Ethtool”](#) (s1-ethtool.html) for more **ethtool** options.

HOTPLUG=*answer*

where *answer* is one of the following:

- › **yes** — This device should be activated when it is hot-plugged (this is the default option).
- › **no** — This device should *not* be activated when it is hot-plugged.

The `HOTPLUG=no` option can be used to prevent a channel bonding interface from being activated when a bonding kernel module is loaded.

See [Section 11.2.4, “Channel Bonding Interfaces”](#) (s2-networkscripts-interfaces-chan.html) for more information about channel bonding interfaces.

HWADDR=*MAC-address*

where *MAC-address* is the hardware address of the Ethernet device in the form `AA:BB:CC:DD:EE:FF`. This directive must be used in machines containing more than one NIC to ensure that the interfaces are assigned the correct device names regardless of the configured load order for each NIC's module. This directive should **not** be used in conjunction with `MACADDR`.



Note

- › Persistent device names are now handled by `/etc/udev/rules.d/70-persistent-net.rules`.
- › `HWADDR` must not be used with System z network devices.
- › See Section 25.3.3, “Mapping subchannels and network device names”, in the *Red Hat Enterprise Linux 6 Installation Guide* (https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Installation_Guide/index.html).

IPADDR=*address*

where *address* is the **IPv4** address.

IPV6ADDR=*address*

where *address* is the first static, or primary, **IPv6** address on an interface.

The format is Address/Prefix-length. If no prefix length is specified, `/64` is assumed. Note that this setting depends on `IPV6INIT` being enabled.

IPV6ADDR_SECONDARIES=*address*

where *address* is one or more, space separated, additional **IPv6** addresses.

The format is Address/Prefix-length. If no prefix length is specified, `/64` is assumed. Note that this setting depends on `IPV6INIT` being enabled.

IPV6INIT=*answer*

where *answer* is one of the following:

- › **yes** — Initialize this interface for **IPv6** addressing.

- › **no** — Do not initialize this interface for **IPv6** addressing. This is the default value.

This setting is required for **IPv6** static and **DHCP** assignment of **IPv6** addresses. It does not affect *IPv6 Stateless Address Autoconfiguration* (SLAAC) as per *RFC 4862* (<http://www.rfc-editor.org/info/rfc4862>).

See [Section D.1.13, “/etc/sysconfig/network”](#) (ch-The_sysconfig_Directory.html#s2-sysconfig-network) for information on disabling **IPv6**.

IPv6_AUTOCONF=answer

where *answer* is one of the following:

- › **yes** — Enable **IPv6** autoconf configuration for this interface.
- › **no** — Disable **IPv6** autoconf configuration for this interface.

If enabled, an **IPv6** address will be requested using *Neighbor Discovery* (ND) from a router running the **radvd** daemon.

Note that the default value of **IPv6_AUTOCONF** depends on **IPv6FORWARDING** as follows:

- › If **IPv6FORWARDING=yes**, then **IPv6_AUTOCONF** will default to **no**.
- › If **IPv6FORWARDING=no**, then **IPv6_AUTOCONF** will default to **yes** and **IPv6_ROUTER** has no effect.

IPv6_MTU=value

where *value* is an optional dedicated MTU for this interface.

IPv6_PRIVACY=rfc3041

where *rfc3041* optionally sets this interface to support *RFC 3041 Privacy Extensions for Stateless Address Autoconfiguration in IPv6* (<http://www.rfc-editor.org/info/rfc3041>). Note that this setting depends on **IPv6INIT** option being enabled.

The default is for *RFC 3041* support to be disabled. Stateless Autoconfiguration will derive addresses based on the MAC address, when available, using the modified **eui-64** method. The address is appended to a prefix but as the address is normally derived from the MAC address it is globally unique even when the prefix changes. In the case of a link-local address the prefix is **fe80::/64** as per *RFC 2462 IPv6 Stateless Address Autoconfiguration* (<http://www.rfc-editor.org/info/rfc2462>).

LINKDELAY=time

where *time* is the number of seconds to wait for link negotiation before configuring the device. The default is 5 secs. Delays in link negotiation, caused by **STP** for example, can be overcome by increasing this value.

MACADDR=MAC-address

where *MAC-address* is the hardware address of the Ethernet device in the form *AA:BB:CC:DD:EE:FF*.

This directive is used to assign a MAC address to an interface, overriding the one assigned to the physical NIC. This directive should **not** be used in conjunction with the **HWADDR** directive.

MASTER=bond-interface

where *bond-interface* is the channel bonding interface to which the Ethernet interface is linked.

This directive is used in conjunction with the **SLAVE** directive.

See [Section 11.2.4, “Channel Bonding Interfaces”](#) (s2-networkscripts-interfaces-chan.html) for more information about channel bonding interfaces.

NETMASK=mask

where *mask* is the netmask value.

NETWORK=*address*

where *address* is the network address. This directive is deprecated, as the value is calculated automatically with `ipcalc`.

NM_CONTROLLED=*answer*

where *answer* is one of the following:

- › **yes** — **NetworkManager** is permitted to configure this device. This is the default behavior and can be omitted.
- › **no** — **NetworkManager** is not permitted to configure this device.



Note

The `NM_CONTROLLED` directive is now, as of Red Hat Enterprise Linux 6.3, dependent on the `NM_BOND_VLAN_ENABLED` directive in `/etc/sysconfig/network`. If and only if that directive is present and is one of **yes**, **y**, or **true**, will **NetworkManager** detect and manage bonding and VLAN interfaces.

ONBOOT=*answer*

where *answer* is one of the following:
`CUSTOMER` (<https://access.redhat.com/>)
`PORTAL`

- › **yes** — This device should be activated at boot-time.
- › **no** — This device should not be activated at boot-time.

PEERDNS=*answer*

where *answer* is one of the following:

- › **yes** — Modify `/etc/resolv.conf` if the `DNS` directive is set, if using `DHCP`, or if using Microsoft's *RFC 1877* (<http://www.rfc-editor.org/info/rfc1877>) `IPCP` extensions with `PPP`. In all cases **yes** is the default.
- › **no** — Do not modify `/etc/resolv.conf`.

SLAVE=*answer*

where *answer* is one of the following:

- › **yes** — This device is controlled by the channel bonding interface specified in the `MASTER` directive.
- › **no** — This device is *not* controlled by the channel bonding interface specified in the `MASTER` directive.

This directive is used in conjunction with the `MASTER` directive.

See [Section 11.2.4, “Channel Bonding Interfaces”](#) (`s2-networkscripts-interfaces-chan.html`) for more about channel bonding interfaces.

SRCADDR=*address*

where *address* is the specified source `IP` address for outgoing packets.

USERCTL=*answer*

where *answer* is one of the following:

- » **yes** — Non-**root** users are allowed to control this device.
- » **no** — Non-**root** users are not allowed to control this device.

Privacy Policy (<http://www.redhat.com/en/about/privacy-policy>)

Customer Portal Terms of Use (<https://access.redhat.com/help/terms/>)

All Policies and Guidelines (<http://www.redhat.com/en/about/all-policies-guidelines>)

Copyright © 2016 Red Hat, Inc.

CUSTOMER (<https://access.redhat.com/>)
PORTAL

