How to assign a specific network interface name in RHEL7, 8

SOLUTION VERIFIED - Updated September 6 2021 at 3:54 AM - English

Environment

- Red Hat Enterprise Linux 7
- Red Hat Enterprise Linux 8

Issue

- Need to directly control network interface names in RHEL7, 8
- How to make NIC names persistent across reboots in Red Hat Enterprise Linux 7, 8

Resolution

This will not consistently work when trying to assign ethX style names. It is not expected to.
This is explained in the following article: ethX support in systemd

RHEL 7 should already provide consistent device naming. Please see the RHEL 7 Networking Guide which explains this feature in detail: CHAPTER 11. CONSISTENT NETWORK DEVICE NAMING

For RHEL 8 please see the RHEL 8 documentation: Chapter 2. Consistent network interface device naming

The following steps override the built-in Consistent Network Device Naming scheme.

To set a custom device name with NetworkManager for an existing connection profile please do the following:

- Ensure the existing connection profile has stored the MAC address of the physical interface it is associated with. In the example below, the connection profile is named "Wired connection 1":

```
# nmcli connection show "Wired connection 1" | grep 802-3-ethernet.mac-address:
802-3-ethernet.mac-address:             --
```

- If the MAC address is missing, add it. First find the MAC address with the `ip link` command and then use the `nmcli` command to modify the connection profile:

```
# ip link show eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
mode DEFAULT qlen 1000
    link/ether xx:xx:xx:xx:xx:xx brd ff:ff:ff:ff:ff:ff

# nmcli connection modify "Wired connection 1" 802-3-ethernet.mac-address
"xx:xx:xx:xx:xx:xx"
```

- Now modify the connection profile's `connection.interface-name` property to set the desired device name. In this example below, the device is named "external-2":

```
# nmcli connection modify "Wired connection 1" connection.interface-name
"external-2"
```

- Reboot the system

- Verify the device has been named as expected:

```
# ip link
3: external-2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state
UP mode DEFAULT qlen 1000
    link/ether xx:xx:xx:xx:xx:xx brd ff:ff:ff:ff:ff:ff
```

To set a custom device name with NetworkManager for an new interface which does not have an existing connection profile please do the following:

- Note the MAC address of the device using the `ip link` command. In the following example, eth0 is used and the MAC address is seen to be xx:xx:xx:xx:xx:xx:

```
# ip link show eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
mode DEFAULT qlen 1000
    link/ether xx:xx:xx:xx:xx:xx brd ff:ff:ff:ff:ff:ff
```

- Use the `nmcli` command to create a new connection profile for eth0. Be sure to specify the MAC address. In the following example, the network device with the specified MAC address will be renamed to "internal-1" upon rebooting the system:

```
# nmcli connection add type ethernet mac "xx:xx:xx:xx:xx:xx" ifname "internal-1"
```

- Reboot the system
- Verify the device name is now "internal-1"

```
# ip link
2: internal-1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state
UP mode DEFAULT qlen 1000
    link/ether xx:xx:xx:xx:xx:xx brd ff:ff:ff:ff:ff:ff
```

To set a custom device name without using NetworkManager:

- In the interface's current interface configuration file (/etc/sysconfig/network-scripts/ifcfg-* file), ensure both the DEVICE and HWADDR parameters are properly set. DEVICE is the name to be given to the interface with the MAC address equal to HWADDR.

Alternatively, setting a custom interface name with a udev rule is still possible.

- Create a udev rules file in the /etc/udev/rules.d/ directory
- The rule can match against the device MAC address or PCI bus address:

```
# MAC address match. Will name the device with the specified MAC address the
value given in the NAME property:
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="xx:xx:xx:xx:xx:xx", ATTR{type}=="1", KERNEL=="*", NAME="myeth2"

# PCI bus address match. Bus address can be seen in the output of the lspci
command.
SUBSYSTEM=="net", ACTION=="add", KERNEL=="eth*", SUBSYSTEMS=="pci",
KERNELS=="0000:00:03.0", NAME="private-3"
```

NOTE: By default `/etc/sysconfig/network` file should be empty for RHEL 7 and above but if there are some contents present inside of the same file, ensure to verify them to match with the expected values in order to avoid any unintended networking issues.

Root Cause

- The `/usr/lib/udev/rules.d/60-net.rules` udev rule file will check each ifcfg file for the presence of the DEVICE and HWADDR parameters. If the interface in question has a MAC address which matches the value of the HWADDR property then it will be assigned the name given by the value of the DEVICE property.
- This feature will not correctly handle setting ethX style names. Please see the `Root Cause` section of the following article for more details:
  - Is it safe to set net.ifnames=0 in RHEL7?

Product(s) **Red Hat Enterprise Linux**

Component **iproute** **kernel** **NetworkManager** **systemd** **Category** **Configure**

Tags **configuration** **networking** **NetworkManager** **rhel_7** **systemd** **udev**

This solution is part of Red Hat's fast-track publication program, providing a huge library of solutions that Red Hat engineers have created while supporting our customers. To give you the knowledge you need the instant it becomes available, these articles may be presented in a raw and unedited form.

People who viewed this solution also viewed

How do I name my network devices based on PCI ID instead of MAC address?

Solution - 2013年8月4日

Why MAC address has changed after reboot?

Solution - 2018年11月5日

Where does a bonding device get its MAC address from?

Solution - 2013年10月7日

5 Comments

NEWBIE

12 Points

22 March 2016 1:12 PM

Reda BENABED

The second procedure don't work fine.

To set a custom device name ....

Reply

COMMUNITY MEMBER

36 Points

8 February 2018 3:19 PM

chunda shi

I think second example not exactly (udev example) It is best to add the configuration of ifcfg-myeth2 and including all contents, in particular including "NAME","DEVICE","HWADDR"

Reply

COMMUNITY MEMBER

73 Points

25 May 2018 10:31 AM

Kasra Amirsarvari

Using NetworkManager is a good approach. Modifying an existing connection or adding a new connection with the correct mac address works fine. But keep in mind, if you use these two options (solution 1 or 2) you will need to keep your connection in tact during the lifecycle. If somehow the NetworkManager connection is removed/deleted the network device will be renamed back to it's default fallback name after a reboot. Be aware of this behavior as it does not keep a persistent name after renaming it.

Reply

COMMUNITY MEMBER

73 Points

25 May 2018 10:50 AM

Kasra Amirsarvari

As a side note... use a "udevadm" command to retrieve all possible device names the naming convention (schemes) follows.

1) Retrieve your existing network interface devices:

```
$ ls -la /sys/class/net
```

This will show you a list of devices (with their current name as known to the system since the last boot).

2) Retrieve device information per interface (some examples you may encounter):

```
$ udevadm info /sys/class/net/eno1
$ udevadm info /sys/class/net/ens1p0
$ udevadm info /sys/class/net/enp8s0f0
$ udevadm info /sys/class/net/eth0
```

A good source for more information can be found at: predictable device names

If you are looking for detailed information use these tools:
- dmidecode - Native installed on a minimal RHEL7 install.
- lshw - Requires a "yum install lshw" on a minimal RHEL7 install.

Reply

NEWBIE

14 Points

7 October 2020 12:26 PM

Mike Howard

IBM/Redhat Fail again.. Did you even QA this.. See that layoffs do,,

Reply