Change default network name (ens33) to old “eth0” on Ubuntu 18.04 / Ubuntu 16.04

By **Raj** Last updated **May 18, 2018**

Just after the Ubuntu installation, I came to know that the network interface name got changed to ens33 from old school eth0.

READ: **Install Ubuntu 18.04 LTS (Bionic Beaver) on UEFI and Legacy BIOS System**

READ: **Install Ubuntu 16.04 LTS (Xenial Xerus) – Step by Step Guide with Screenshots**

If you ever interested in changing interface names to old type ethX, here is the tutorial for you.

As you can see in the following command, my system is having a network adapter called ens33.

This is just the case of VMware environment, it may vary depends on the hardware but the steps to get back ethX will be the same.

```
$ ip a

1: lo: <loopback,up,lower_up> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: </loopback,up,lower_up>ens33: <broadcast,multicast,up,lower_up> mtu 1500 qdisc
    link/ether 00:0c:29:05:a3:e2 brd ff:ff:ff:ff:ff:ff
    </broadcast,multicast,up,lower_up>inet 192.168.12.12/24 brd 192.168.12.255 sc
       valid_lft 1683sec preferred_lft 1683sec
    inet6 fe80::20c:29ff:fe05:a3e2/64 scope link
       valid_lft forever preferred_lft forever
```

From the dmesg command, you can see that the device got renamed during the system boot.

```
$ dmesg | grep -i eth

[    3.050064] e1000 0000:02:01.0 eth0: (PCI:66MHz:32-bit) 00:0c:29:05:a3:e2
[    3.050074] e1000 0000:02:01.0 eth0: Intel(R) PRO/1000 Network Connection
[    3.057410] e1000 0000:02:01.0 ens33: renamed from eth0
```

To get an ethX back, edit the grub file.

```
$ sudo nano /etc/default/grub
```

Look for "**GRUB_CMDLINE_LINUX**" and add the following"**net.ifnames=0 biosdevname=0**".

From:

```
GRUB_CMDLINE_LINUX=""
```

To:

```
GRUB_CMDLINE_LINUX="net.ifnames=0 biosdevname=0"
```

Generate a new grub file using the following command.

```
$ sudo grub-mkconfig -o /boot/grub/grub.cfg

Generating grub configuration file ...
Warning: Setting GRUB_TIMEOUT to a non-zero value when GRUB_HIDDEN_TIMEOUT is set
Found linux image: /boot/vmlinuz-4.4.0-15-generic
Found initrd image: /boot/initrd.img-4.4.0-15-generic
Found memtest86+ image: /memtest86+.elf
Found memtest86+ image: /memtest86+.bin
done
```

Edit the interface file and change the network device name so that you will have a DHCP or static IP address for ethX.

READ: **Install and configure DHCP server on CentOS 7 / Ubuntu 16.04 / Debian 9**

Assign IP Address

For Ubuntu 18.04

You can assign an IP address to the system using **netplan** – a new network configuration tool.

READ: **How To Configure Static IP Address in Ubuntu 18.04 using Netplan**

For Ubuntu 16.04 / Older

DHCP:

If your infrastructure has a DHCP server and you want to leverage that, then:

```
$ sudo nano /etc/network/interfaces
```

Update below lines in **/etc/network/interfaces** files so that the network card can get an IP address from DHCP server.

FROM:

```
auto ens33
iface ens33 inet dhcp
```

TO:

```
auto eth0
iface eth0 inet dhcp
```

Static:

If your infrastructure does not have a DHCP server, then you will need to configure a static IP address for all network interfaces on your Ubuntu machine.

```
$ sudo nano /etc/network/interfaces
```

From:

```
auto ens33
iface ens33 inet static
           address 192.168.12.12
           netmask 255.255.255.0
           dns-nameservers 192.168.12.2
           gateway 192.168.12.2
```

To:

```
auto eth0
iface eth0 inet static
           address 192.168.12.12
           netmask 255.255.255.0
           dns-nameservers 192.168.12.2
           gateway 192.168.12.2
```

Reboot your system.

```
$ sudo reboot
```

After the system reboot, just check whether you have an ethX back.

```
$ ip a
1: lo: <loopback,up,lower_up> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: </loopback,up,lower_up>eth0: <broadcast,multicast,up,lower_up> mtu 1500 qdisc
    link/ether 00:0c:29:05:a3:e2 brd ff:ff:ff:ff:ff:ff
    </broadcast,multicast,up,lower_up>inet 192.168.12.12/24 brd 192.168.12.255 sc
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe05:a3e2/64 scope link
       valid_lft forever preferred_lft forever
```

That’s All.