

네트워크 관련 설정 방법

Selecting Network Configuration Methods

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/networking_guide/ch-configuring_ip_networking#sec-Selecting_Network_Configuration_Methods

- To configure a network interface using **NetworkManager**, use one of the following tools:
 - the text user interface tool, **nmtui**. For more details, see [Section 3.2, “Configuring IP Networking with nmtui”](#).
 - the command-line tool, **nmcli**. For more details, see [Section 3.3, “Configuring IP Networking with nmcli”](#).
 - the graphical user interface tools, **GNOME GUI**. For more details, see [Section 3.4, “Configuring IP Networking with GNOME GUI”](#).
- To configure a network interface **without** using **NetworkManager**:
 - edit the `ifcfg` files manually. For more details, see [Section 3.5, “Configuring Ip Networking with ifcfg Files”](#).
 - use the `ip` commands. This can be used to assign IP addresses to an interface, but changes are not persistent across reboots; when you reboot, you will lose any changes. For more details, see [Section 3.6, “Configuring IP Networking with ip Commands”](#).
- To configure the network settings when the root filesystem is **not** local:
 - use the kernel command-line. For more details, see [Section 3.7, “Configuring IP Networking from the Kernel Command line”](#).

Applications or Tool	Description
NetworkManager	기본적인 네트워크 데몬
nmtui	NetworkManager 를 위한 curses 를 기반으로 하는 간단한 텍스트 유저 인터페이스(TUI)
nmcli	유저들과 스크립트들이 NetworkManager 와 상호 통신하도록 제공되는 커맨드 라인 툴
control-center	GNOME 셸에서 제공되는 GUI 툴
nm-connection-editor	control-center 에 아직 조정되지 않은 어떠한 작업들을 위해 사용할 수 있는 GTK+3 어플리케이션

NetworkManager 소개

RHEL7에서는, 기본적인 네트워킹 서비스는 NetworkManager에 의해 제공된다. NetworkManager는 동적인 네트워크 컨트롤 및 설정 데몬으로, 네트워크 장치들과 연결이 사용 가능할 때 연결하고 동작하도록 해주는 역할을 한다.

리눅스에서의 네트워크 장치 이름

- 랜 카드(NIC)가 리눅스에 장착되었을 때 자동으로 이 장치의 이름은 ens32 또는 ens33으로 인식한다.
- 이 랜카드의 이름은 앞으로 네트워크 정보를 파악하거나, 네트워크를 정지 또는 가동할 때 자주 사용한다.
- 네트워크 장치의 이름은 VMware 또는 진짜 컴퓨터, 어느 장비에 설치하냐에 따라서 ens32, ens33, eth0, eth1 등 다양한 형태로 인식하기도 하며, VMware 버전에 따라서 다른 이름으로 인식하기도 한다.

네트워크 관련 명령어

`ifconfig [-v] [-a] [-s] [interface]`

- 해당 장치의 IP 주소와 관련 정보를 출력해주는 명령어이다.

`nslookup`

- DNS 서버의 작동을 테스트하는 명령어이다.

`ping IP 주소`

- 해당 컴퓨터가 네트워크상에서 응답하는지를 테스트하는 간편한 명령어이다.

네트워크 설정과 관련된 주요 파일

`/etc/sysconfig/network-scripts/ifcfg-<장치명>`

- 시스템관리자는 수동으로 네트워크 인터페이스를 구성하여 ifcfg 파일을 편집할 수 있다.

- ifcfg(인터페이스 구성)파일은 개별 네트워크 장치의 소프트웨어 인터페이스를 제어한다. 시스템이 부팅할 때 이 파일을 사용하여 가져올 인터페이스와 구성 방법을 결정한다.
- 일반적으로 이 파일의 접미어 이름은 구성 파일이 제어하는 장치의 이름으로 지정된다.
- 파일의 위치 : /etc/sysconfig/network-scripts/ifcfg-<장치명>
- ipcalc 가 자동으로 계산하므로 네트워크 또는 브로드 캐스트 주소를 지정할 필요가 없다.

/etc/resolv.conf

- DNS 서버의 정보와 호스트 이름이 들어 있는 파일이다.
- 임시로 사용되는 파일이며 네트워크가 재시작되면 다시 내용이 초기화된다.

/etc/hosts

- 현 컴퓨터의 호스트 이름이 들어있는 파일이다.

네트워크 설정하기

네트워크 설정 전 ifcfg 스크립트 확인

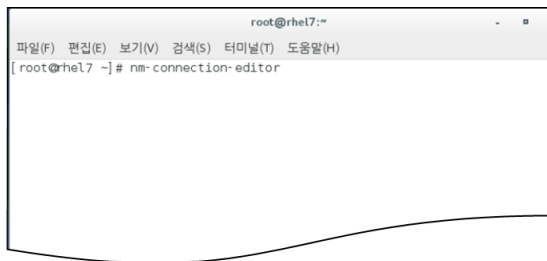
1. /etc/sysconfig/network-scripts/ifcfg-ens33 파일 내용 확인

```
#cat /etc/sysconfig/network-scripts/ifcfg-ens33
```

```
TYPE=Ethernet
BOOTPROTO=dhcp
DEFROUTE=yes
PEERDNS=yes
PEERROUTES=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_PEERDNS=yes
IPV6_PEERROUTES=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=ens33
UUID=c75a7b39-ceb5-42e3-984e-1d62c127177d
DEVICE=ens33
ONBOOT=no
```

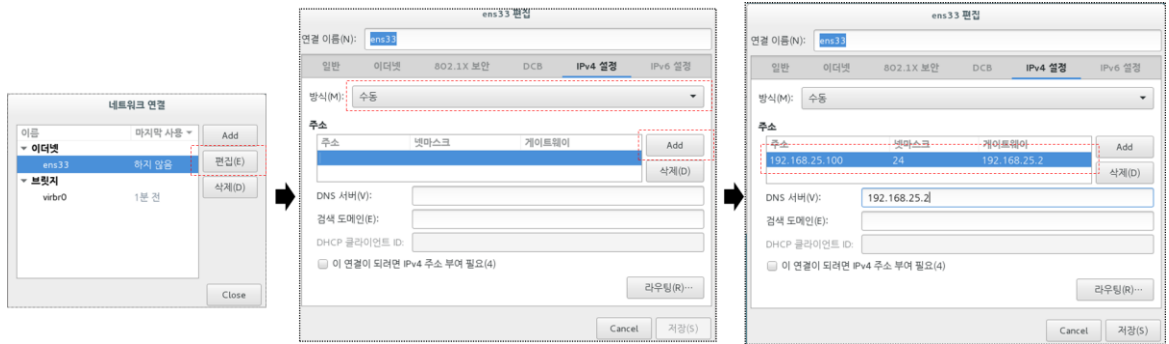
네트워크 설정하기 1. nm-connection-editor 로 네트워크 설정하기

1. nm-connection-editor 명령을 입력하여 실행



2. [네트워크 연결] 창이 나오면, 장치이름 클릭 → [편집] → [방식] 수동 → [주소] 부분 [Add] 클릭하여 IP 주소, 서브넷마스크, 게이트웨이, DNS 서버 주소를 다음과 같이 입력→[저장]

IP 주소	192.168.25.100
서브넷 마스크	255.255.255.0
게이트웨이	192.168.25.2
DNS 서버	192.168.25.2



3. 네트워크 설정 파일 확인

```
# cat /etc/sysconfig/network-scripts/ifcfg-ens33
TYPE=Ethernet
BOOTPROTO=none
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=ens33
UUID=c75a7b39-ceb5-42e3-984e-1d62c127177d
DEVICE=ens33
ONBOOT=no
IPADDR=192.168.25.100
PREFIX=24
GATEWAY=192.168.25.2
DNS1=192.168.25.2
IPV6_PEERDNS=yes
IPV6_PEERROUTES=yes
IPV6_PRIVACY=no
```

4. NetworkManager 에게 설정 변경사항을 적용시키기 위해 다음 명령어 수행

```
# nmcli con up ens33
```

연결이 성공적으로 활성화되었습니다 (D-Bus 활성 경로:
/org/freedesktop/NetworkManager/ActiveConnection/3)

5. 확인

```
# ifconfig -a
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
inet 192.168.25.100  netmask 255.255.255.0  broadcast 192.168.25.255
inet6 fe80::38eb:f629:6366:1f  prefixlen 64  scopeid 0x20<link>
ether 00:0c:29:a3:0e:5a  txqueuelen 1000  (Ethernet)
RX packets 28382  bytes 2611882 (2.4 MiB)
RX errors 0  dropped 0  overruns 0  frame 0
TX packets 63  bytes 9631 (9.4 KiB)
TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

- ✓ 만약 /etc/sysconfig/network-scripts/ifcfg-**<장치명>** 파일을 수정했다면 다음 명령어로 load 시켜준다.

```
# nmcli con load /etc/sysconfig/network-scripts/ifcfg-<장치명>
```

6. 인터넷 연결 확인

네트워크 설정하기 2. /etc/sysconfig/network-scripts/ifcfg-<장치명> 파일 수정

이전의 RHEL 릴리즈에서 네트워크를 설정하는 기본 방식은 network scripts 를 사용하여 네트워크를 설정하는 것이었다. network scripts 는 보통 /etc/init.d/network 스크립트와 그것이 호출하는 다른 스크립트들(/etc/sysconfig/network-scripts/*)이다.

1. /etc/sysconfig/network-scripts/ifcfg-ens33 파일에서 IP 를 192.168.25.200 으로 수정
vi /etc/sysconfig/network-scripts/ifcfg-ens33

```
TYPE=Ethernet
BOOTPROTO=none
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=ens33
UUID=51639fde-cab6-4c25-97cc-0d17c8bb5ef4
DEVICE=ens33
ONBOOT=no
DNS1=192.168.25.2
IPADDR=192.168.25.200
PREFIX=24
GATEWAY=192.168.25.2
IPV6_PEERDNS=yes
IPV6_PEERROUTES=yes
IPV6_PRIVACY=no
```

2. RHEL7 에서는 ifcfg 파일을 편집할 때, NetworkManager 는 자동으로 변경사항을 알지 못하기 때문에 변경사항을 알 수 있도록 해주어야 한다. 에디터를 사용하여 설정파일을 수정했다면, NetworkManager 는 해당 설정 파일들을 다시 읽어들여야 하기 때문에 root 로 다음 명령을 수행해야 한다.

```
# nmcli con reload /etc/sysconfig/network-scripts/ifcfg-ens33
```

3. 장치를 다시 활성화 시킨다.

```
# nmcli con up ens33
```

연결이 성공적으로 활성화되었습니다 (D-Bus 활성 경로:

/org/freedesktop/NetworkManager/ActiveConnection/5)

4. 확인

```
# ifconfig -a
```

```
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.25.200  netmask 255.255.255.0  broadcast 192.168.25.255
    inet6 fe80::d158:9eff:9c4c:71ab  prefixlen 64  scopeid 0x20<link>
    ether 00:0c:29:5d:24:8c  txqueuelen 1000  (Ethernet)
    RX packets 8995  bytes 5258070 (5.0 MiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 4307  bytes 527902 (515.5 KiB)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```


네트워크 설정하기 3. nmcli 명령어를 사용하여 네트워크 설정하기

1. 네트워크 연결 확인

1) 네트워크 모든 연결 확인

```
# nmcli con show
```

이름	UUID	유형	장치
ens33	51639fde-cab6-4c25-97cc-0d17c8bb5ef4	802-3-ethernet	ens33
virbr0	21408f21-b5d6-4ad1-9b14-20ee50cc17b7	bridge	virbr0

2) 장치상태 표시

```
# nmcli dev status ens33
```

알려져있지 않은 매개변수입니다: ens33

장치	유형	상태	연결
virbr0	bridge	연결됨	virbr0
ens33	ethernet	연결됨	ens33
lo	loopback	관리되지 않음	--
virbr0-nic	tun	관리되지 않음	--

3) 현재의 장치설정을 확인합니다.

```
# nmcli dev show ens33
```

GENERAL.장치:	ens33
GENERAL.유형:	ethernet
GENERAL.하드웨어주소:	00:0C:29:5D:24:8C
GENERAL.MTU:	1500
GENERAL.상태:	100 (연결됨)
GENERAL.연결 :	ens33
GENERAL.CON-경로:	

```
/org/freedesktop/NetworkManager/ActiveConnection/6
```

WIRED-PROPERTIES.캐리어: 켜짐

IP4.주소[1]: 192.168.25.200/24

IP4.게이트웨이: 192.168.25.2

IP4.DNS[1]: 192.168.25.2

IP6.주소[1]: fe80::d158:9eff:9c4c:71ab/64

IP6.게이트웨이:

2. nmcli 명령어로 IP 변경(192.168.25.200 에서 192.168.25.100 으로 변경)

```
# nmcli con mod "ens33" ip4 192.168.25.100/24 gw4 192.168.25.2
```

3. 장치 활성화

```
# nmcli con up ens33
```

연결이 성공적으로 활성화되었습니다 (D-Bus 활성 경로:
/org/freedesktop/NetworkManager/ActiveConnection/7)

4. 확인

```
# ifconfig -a
```

```
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
```

```
inet 192.168.25.100 netmask 255.255.255.0  broadcast 192.168.25.255
```

```
inet6 fe80::d158:9eff:9c4c:71ab prefixlen 64  scopeid 0x20<link>
```

```
ether 00:0c:29:5d:24:8c  txqueuelen 1000  (Ethernet)
```

```
RX packets 14745  bytes 5871854 (5.5 MiB)
```

```
RX errors 0  dropped 0  overruns 0  frame 0
```

```
TX packets 5606  bytes 666914 (651.2 KiB)
```

```
TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```