마. 실습3 - Multi Node Cluster(완전분산모드)

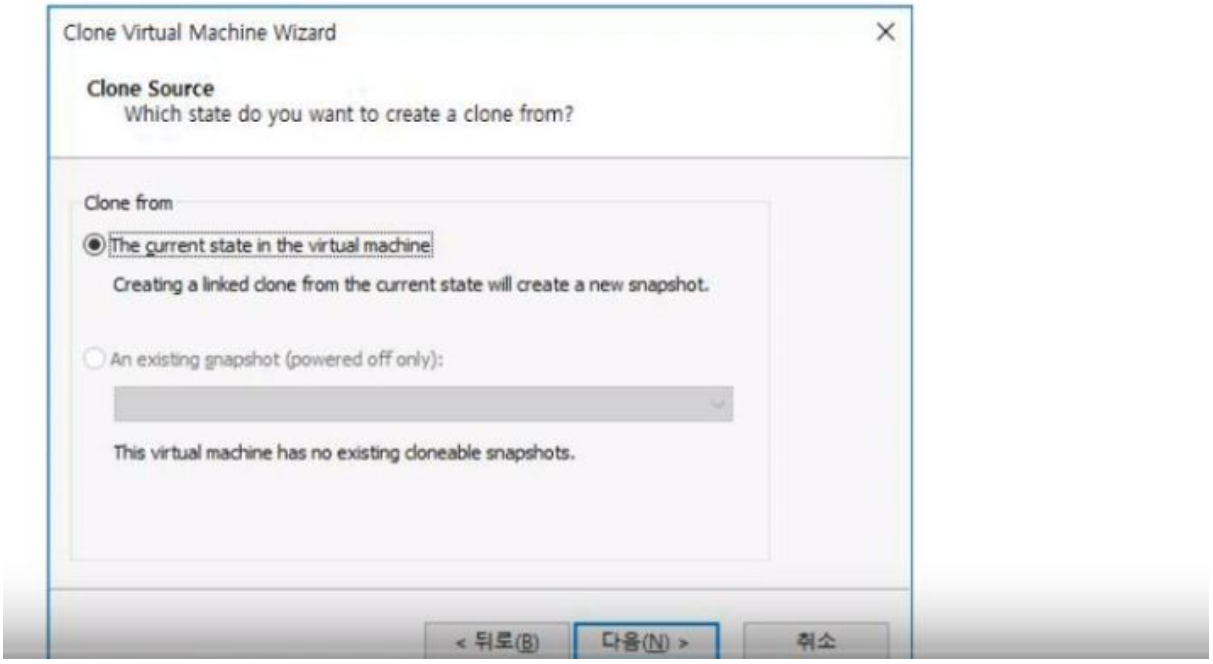
1) VMplayer의 경우 복제 기능이 없으므로 가상머신이 설치된 디렉토리를 복사하여 아래와 같이 디렉토리 구성

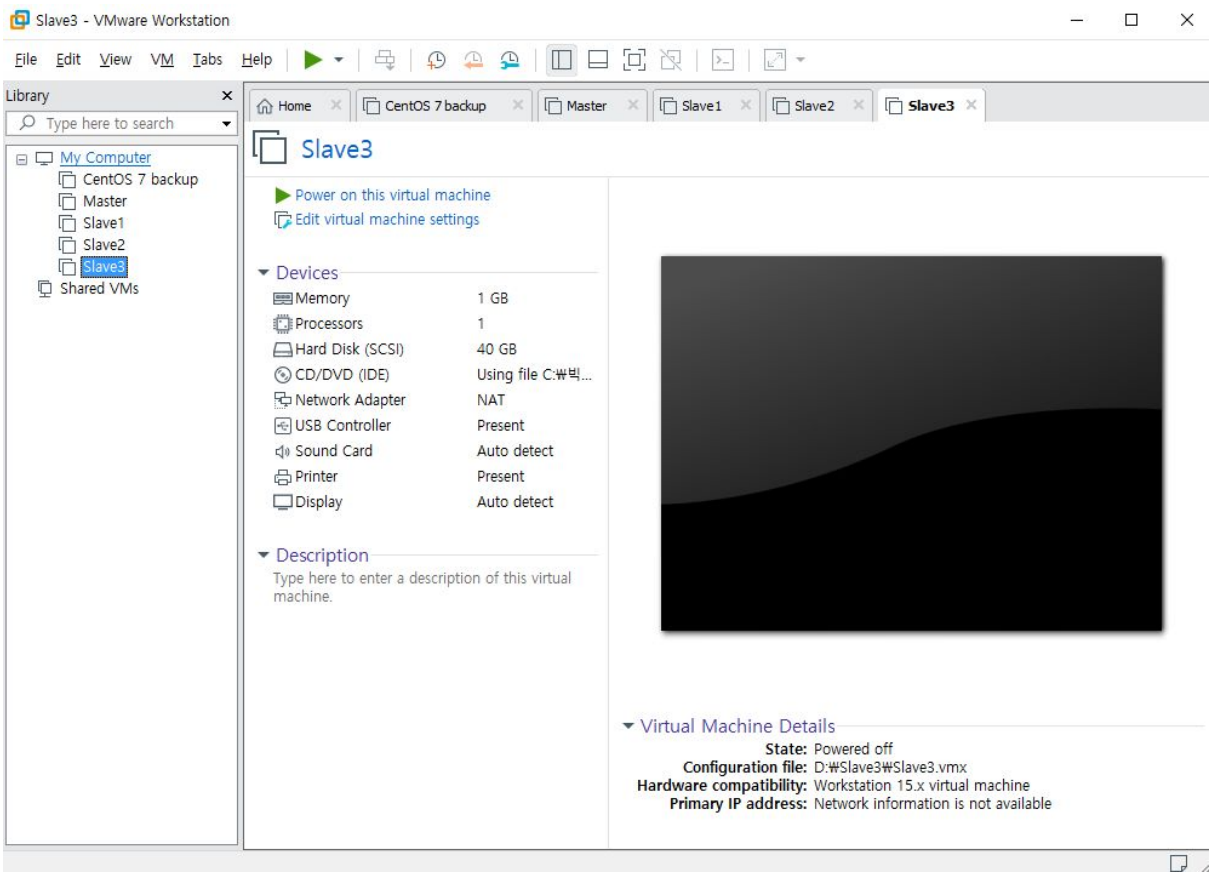
d:₩centos₩backup
d:₩centos₩master
d:₩centos₩slave1
d:₩centos₩slave2
d:₩centos₩slave3

VMplayer버전 같은 경우는

3) VMware Workstation(정식버전)의 가상머신 복제 방법

1대를 설치한 후 vmware에서 clone 복제
Manage – Clone





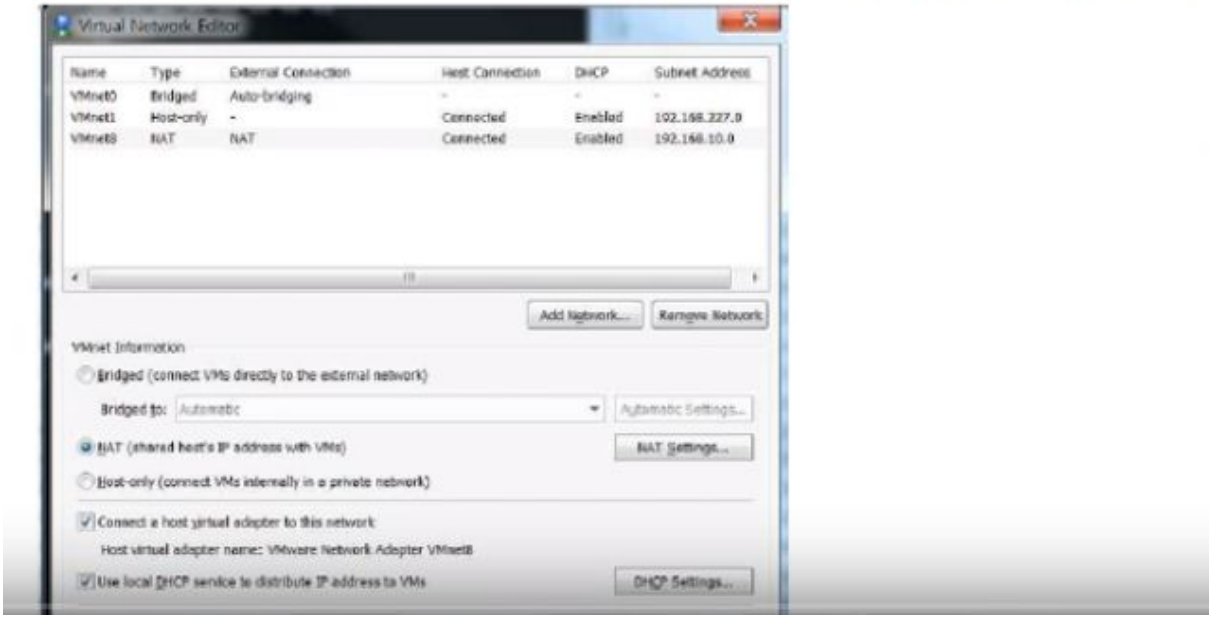
Master를 통해서 slave1,2,3 clone생성한다.

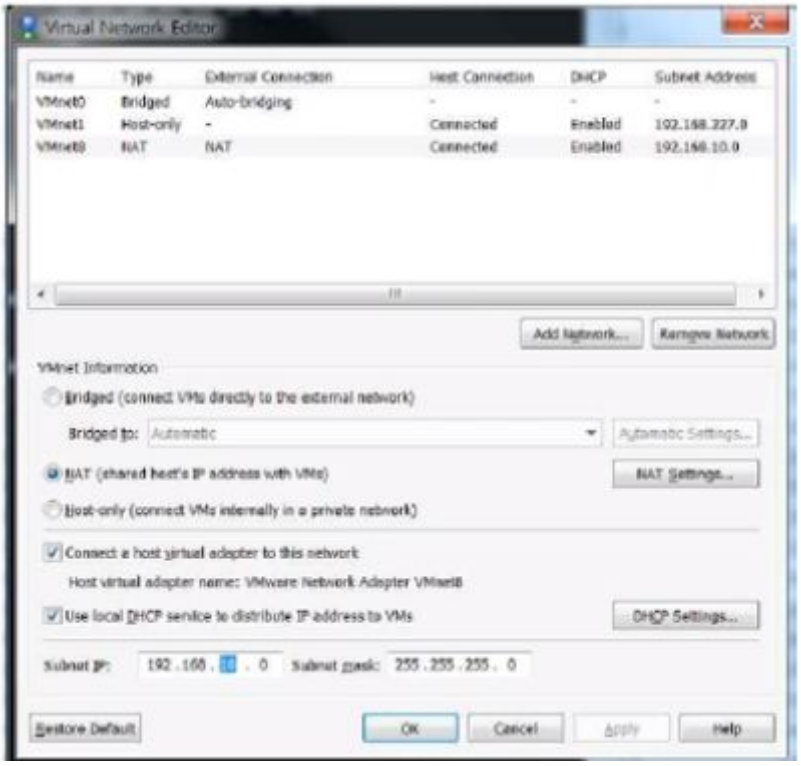
4) 네트워크 설정 작업

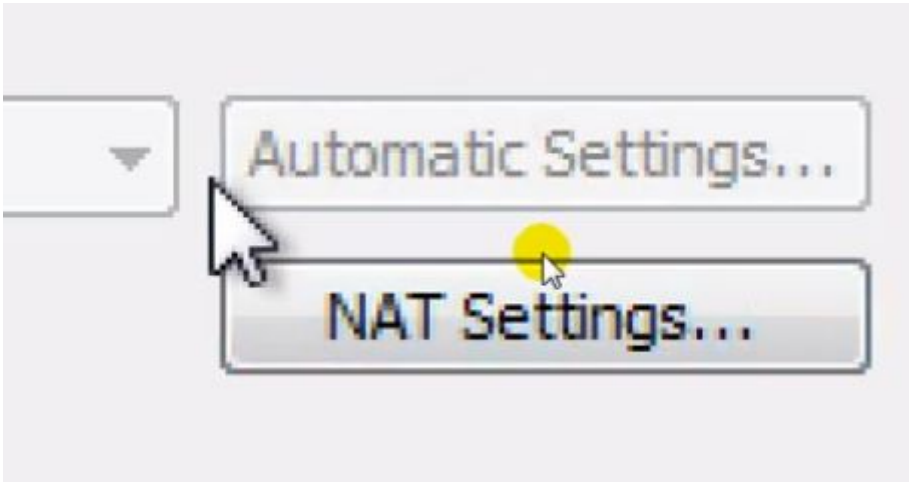
VMware Workstation Pro 버전에서 실행

Edit – Virtual Network Editor

VMnet8을 선택한 후 아래쪽의 Subnet IP에서 3번째 자리를 10으로 변경

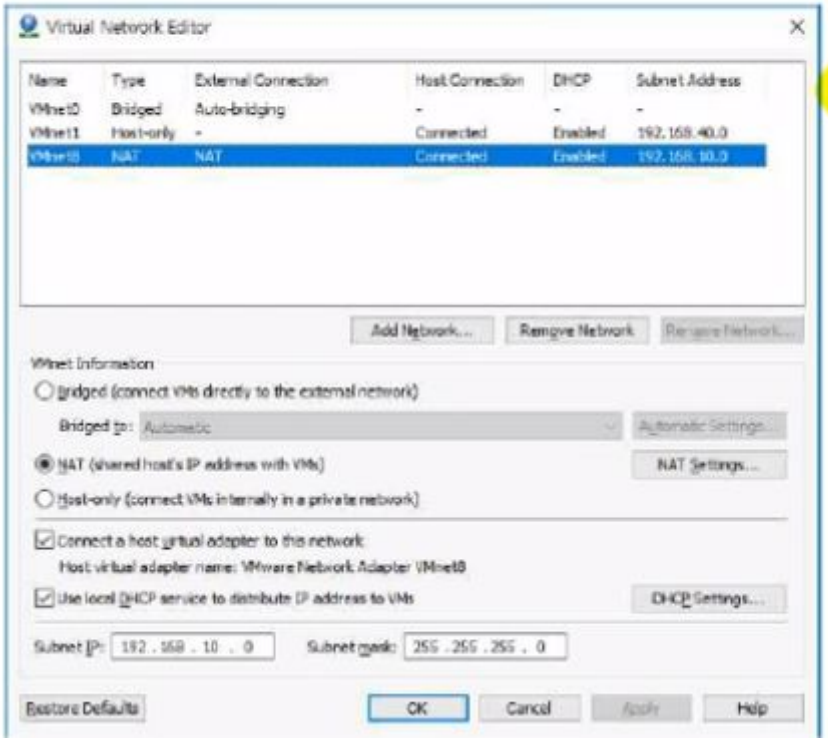






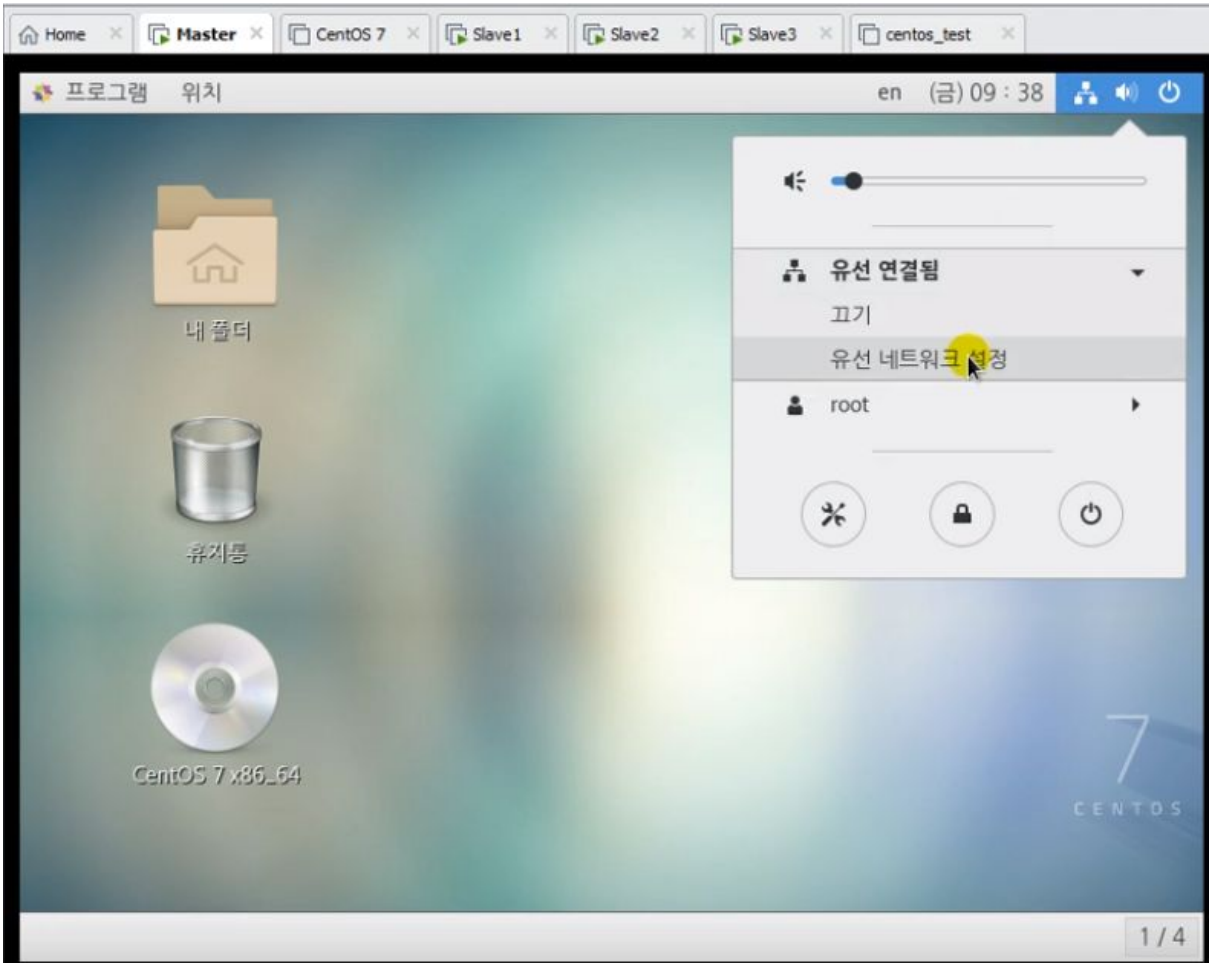
NAT Settings 버튼 클릭

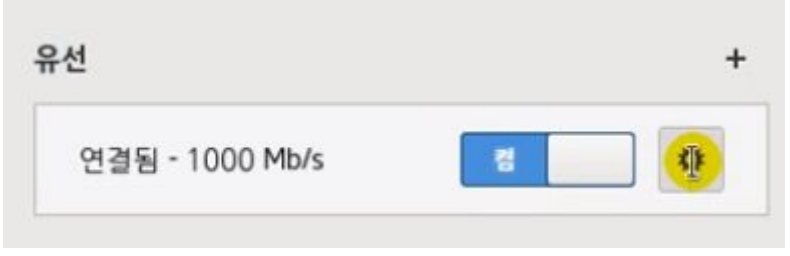
Gateway IP를 192.168.10.2에서 192.168.10.254로 변경

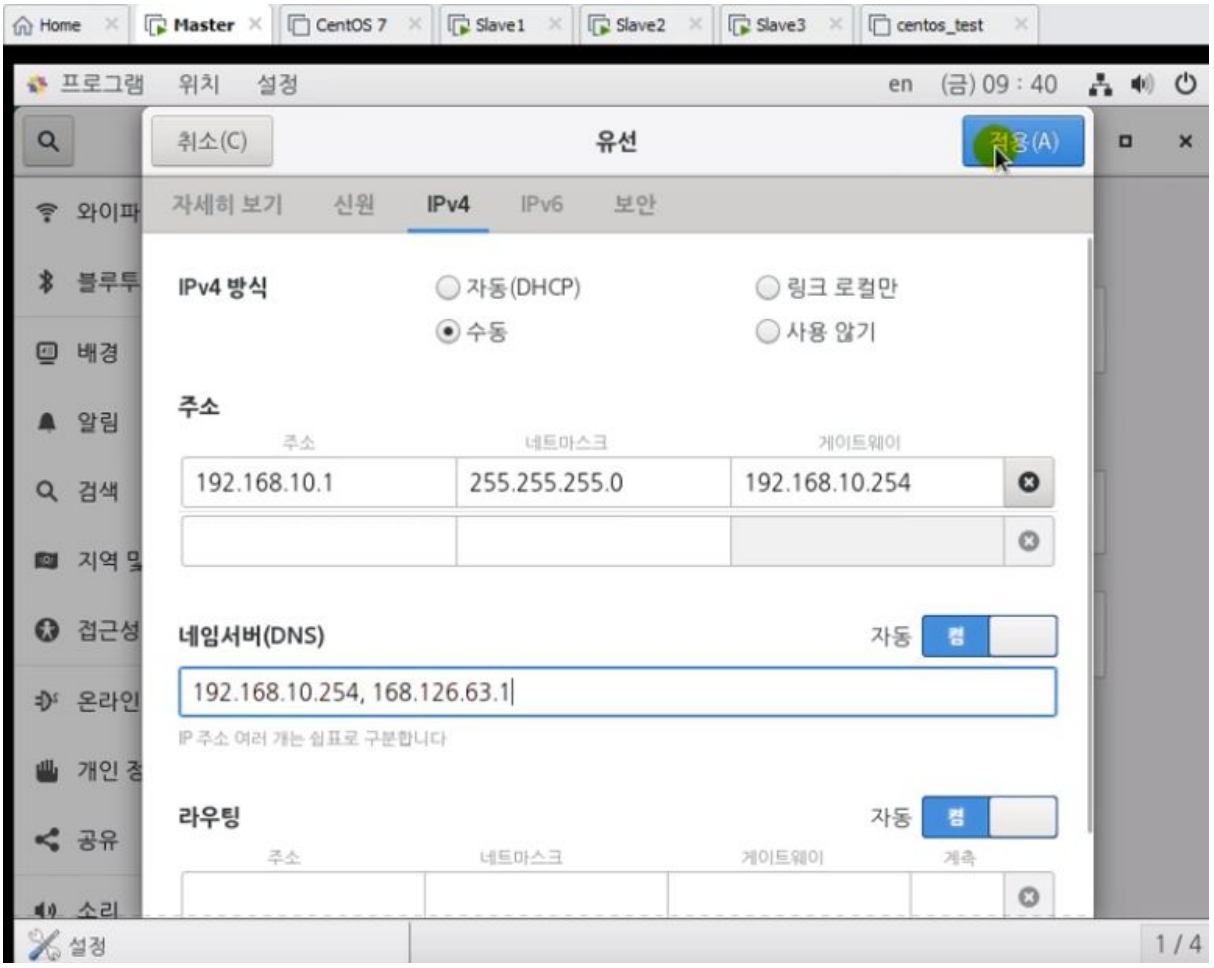


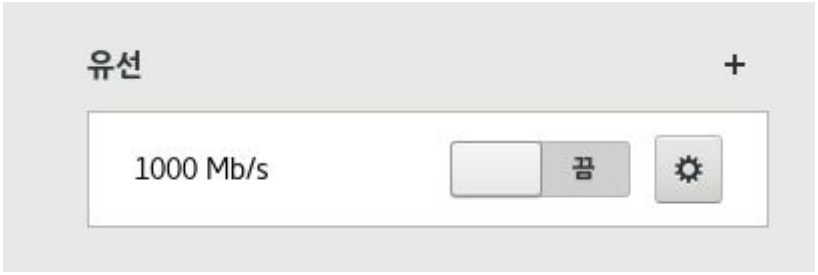
가상 머신 4대를 모두 구동시킨 후 각각 ip를 변경한다.

ip주소
master 192.168.10.1
slave1 192.168.10.2
slave2 192.168.10.3
slave3 192.168.10.4
네트마스크 : 255.255.255.0
게이트웨이 : 192.168.10.254
네임서버 : 192.168.10.254
168.126.63.1





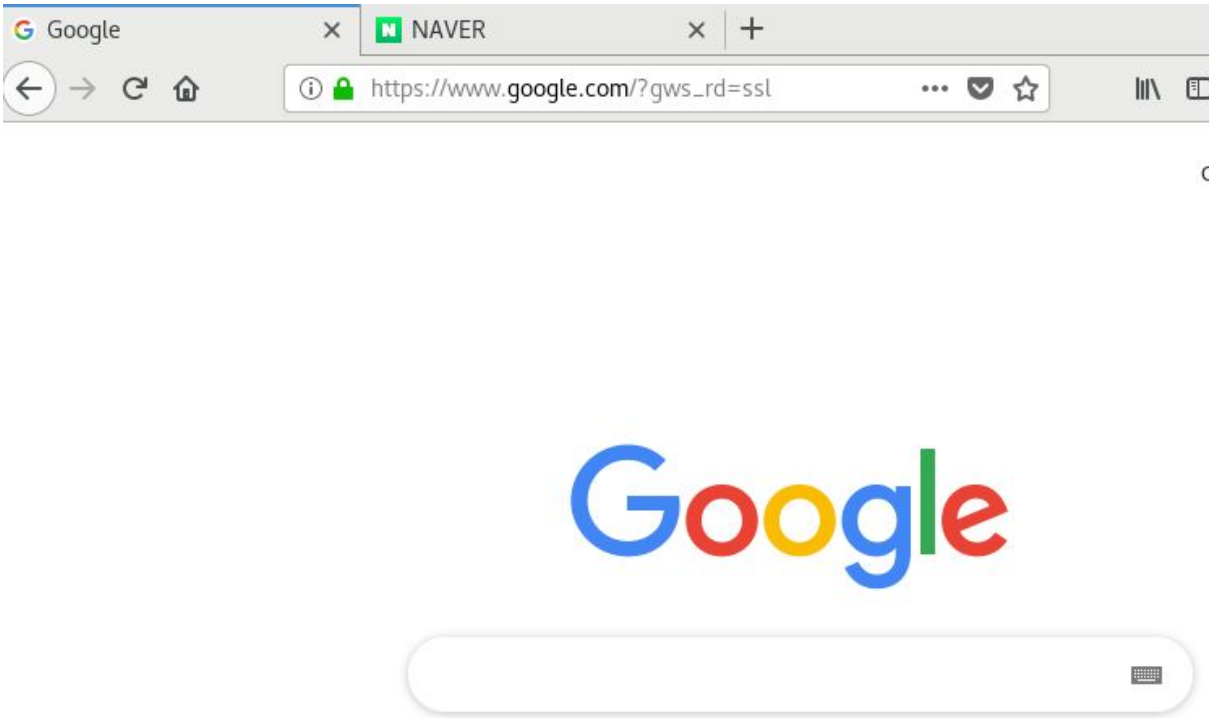




껏다가 켜야 적용된다.

ifconfig, firefox접속을 통해 확인.

```
[root@localhost ~]# ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.10.1  netmask 255.255.255.0  broadcast 192.168.10.255
        inet6 fe80::fc2d:5100:862f:1da3  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:2a:5a:1d  txqueuelen 1000  (Ethernet)
        RX packets 2890  bytes 3342193 (3.1 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 2304  bytes 224941 (219.6 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```



slave1,2,3까지 모두 설정

5) 호스트 설정(모든 노드에서 실행)

```
gedit /etc/hosts
```

```
127.0.0.1 localhost
192.168.10.1 master
192.168.10.2 slave1
192.168.10.3 slave2
192.168.10.4 slave3
```

hosts와 hostname을 일치시키는 작업
master에서 실행

```
gedit /etc/hostname
```

```
master
```

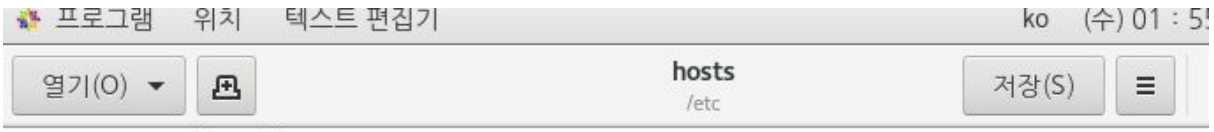
slave1에서 실행

```
gedit /etc/hostname
```

4대 모두 실행

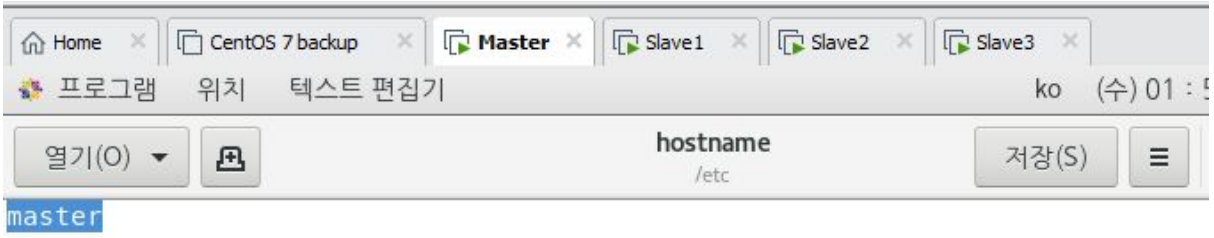
gedit /etc/hostname

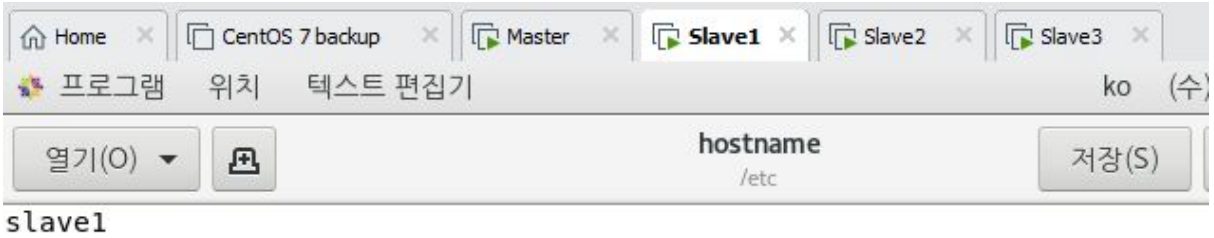
```
[root@localhost ~]# gedit /etc/hosts
```

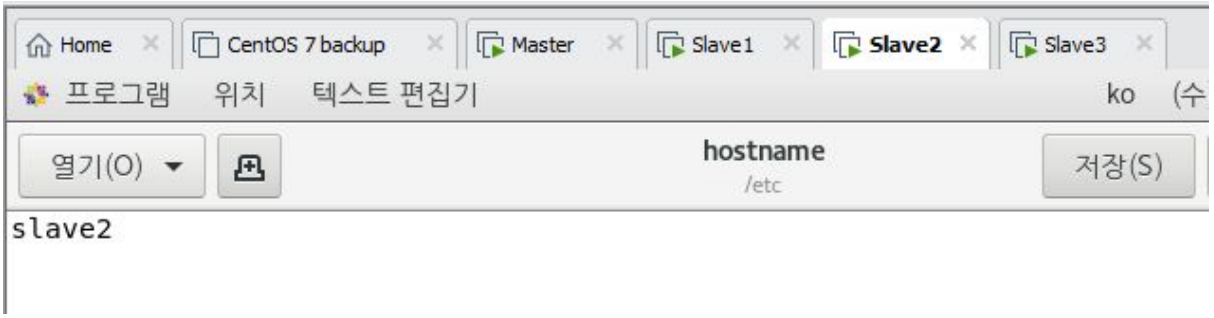


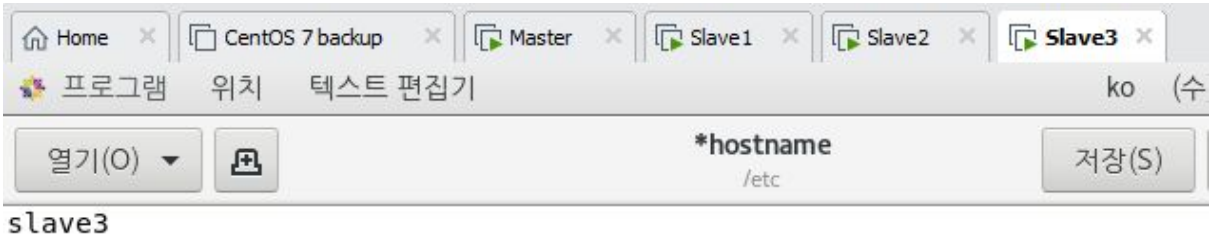

```
127.0.0.1   localhost
192.168.10.1 master
192.168.10.2 slave1
192.168.10.3 slave2
192.168.10.4 slave3
```

```
[root@localhost ~]# gedit /etc/hostname
```

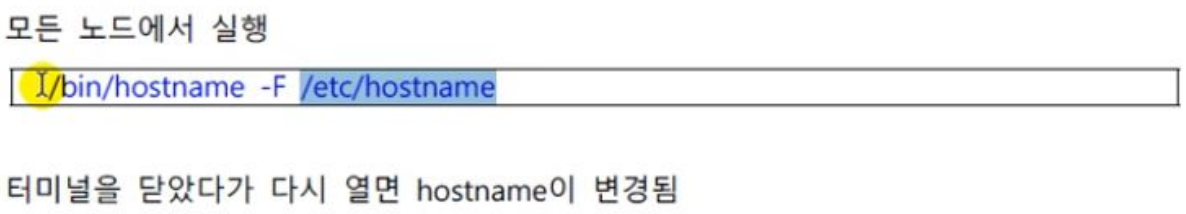




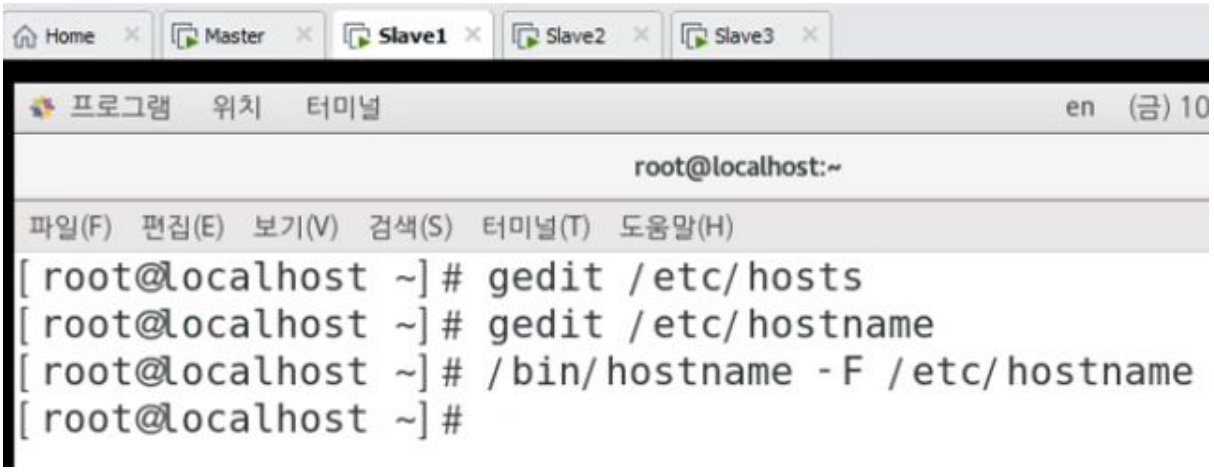




자, 이제 모든 hostname변경된 사항 적용



/bin/hostname -F /etc/hostname



재부팅(모든 노드에서 실행)

```
reboot
```

재부팅 후 hostname이 변경되었는지 확인(모든 노드)

```
hostname
```

ping 테스트

master에서 실행

```
ping slave1
ping slave2
ping slave3
```

slave1에서 실행

```
ping master
ping slave2
ping slave3
```

slave2에서 실행

```
ping master
ping slave1
ping slave3
```

slave3에서 실행

```
ping master
ping slave1
```

```
64 bytes from slave1 (192.168.10.2): icmp_seq=158 ttl=64 time=0.347 ms
64 bytes from slave1 (192.168.10.2): icmp_seq=159 ttl=64 time=0.414 ms
64 bytes from slave1 (192.168.10.2): icmp_seq=160 ttl=64 time=0.391 ms
64 bytes from slave1 (192.168.10.2): icmp_seq=161 ttl=64 time=0.357 ms
64 bytes from slave1 (192.168.10.2): icmp_seq=162 ttl=64 time=0.378 ms
64 bytes from slave1 (192.168.10.2): icmp_seq=163 ttl=64 time=0.484 ms
^C
--- slave1 ping statistics ---
163 packets transmitted, 163 received, 0% packet loss, time 162237ms
rtt min/avg/max/mdev = 0.320/0.900/6.937/0.708 ms
[root@master ~]#
```

Ctrl+C해야 ping테스트가 끝난다.
다음과 같이 했을때, loss가 없다는 것은 잘 연결이 되어있다는 뜻이다.

6) SSH 공개키 복사

로컬 서버 → 원격 서버로 파일 전송

scp [옵션] [원본 경로 및 파일] [계정명]@[원격지IP주소]:[전송할 경로]

scp /home/me/wow.html abc@111.222.333.444:/home/abc/

- master에서 생성한 공개 키를 모든 datanode로 복사(master에서 실행)

```
scp -rp ~/.ssh/authorized_keys root@slave1:~/.ssh/authorized_keys
scp -rp ~/.ssh/authorized_keys root@slave2:~/.ssh/authorized_keys
scp -rp ~/.ssh/authorized_keys root@slave3:~/.ssh/authorized_keys
```

하둡은 분산시스템을 기반으로 작동한다.
즉, 원격 접속을 통해서 데이터를 주고받아야한다.
따라서, 보안 접속이 필요하고 이를 SSH를 사용한다.
이는, 지난 시간에 배웠던 내용이다.

그렇다면, SSH는 공개키와 개인키를 통해서 인증절차를 거치는데,
따라서 공개키를 모두 나눠줘야한다.