

NetEM(Network Emulator) 사용 가이드

2015년 2월 24일

경북대학교 통신프로토콜연구실

김우주 kachukun@gmail.com

요 약

netem은 광역 네트워크의 특성을 에뮬레이팅하는 실험 프로토콜을 위한 네트워크 에뮬레이션 기능을 제공한다. netem은 iproute2에 포함되어있는 기능으로써 2.6 이상의 리눅스 커널을 사용하고 있다면 기본적으로 사용할 수 있도록 되어있다. 해당 문서의 예제는 Ubuntu 12.10 64-bit를 2대의 VMware에 설치하고 Local Network로 연결한 환경에서 작성되었다

목 차

1. (OPTION) 설치하기	2
2. NETWORK INTERFACE 설정하기	2
3. PACKET DELAY (고정값).....	3
4. PACKET DELAY (일정 범위값)	4
5. PACKET LOSS.....	4
6. PACKET DUPLICATION	5
7. PACKET CORRUPTION	6
8. 다중 설정.....	7
참고 문헌.....	7

1. (option) 설치하기

대부분의 경우 iproute가 Linux Kernel에 기본적으로 포함되어있다. 만일 iproute가 없거나 최신 버전으로 업그레이드할 필요가 있는 경우, apt-get 명령어를 이용하여 설치할 수 있다.

```
root@kim-virtual-machine:/home/kim# apt-get install iproute
Reading package lists... Done
Building dependency tree
Reading state information... Done
iproute is already the newest version.
The following package was automatically installed and is no longer required:
  linux-headers-3.5.0-17
Use 'apt-get autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 3 not upgraded.
root@kim-virtual-machine:/home/kim#
```

그림 1. Iproute 설치

2. Network interface 설정하기

*Interface의 설정 값 확인하기.

```
root@kim-virtual-machine:/home/kim# tc qdisc show dev eth0
qdisc pfifo_fast 0: root refcnt 2 bands 3 priomap  1 2 2 2 1 2 0 0  1 1 1 1 1 1 1 1
root@kim-virtual-machine:/home/kim#
```

그림 2. interface 설정 값 확인

아무 설정이 되지 않은 eth0의 설정 값을 확인하는 화면이다. 다른 interface를 확인하고 싶다면 eth0에 다른 interface 이름을 넣어주면 된다.

*Interface에 설정 추가하기

```
root@kim-virtual-machine:/home/kim# tc qdisc add dev eth0 root netem delay 100ms
root@kim-virtual-machine:/home/kim# tc qdisc show dev eth0
qdisc netem 8002: root refcnt 2 limit 1000 delay 100.0ms
```

그림 3. interface 설정 추가

eth0에 100ms의 delay 설정을 추가하고 설정 값을 확인하는 화면이다. interface에 아무런 설정이 되어있지 않는 경우는 add를 사용하여 설정을 추가한다. delay 설정에 관한 자세한 내용은 뒤에 설명한다.

*Interface의 설정 변경하기

```
root@kim-virtual-machine:/home/kim# tc qdisc change dev eth0 root netem delay 50ms
root@kim-virtual-machine:/home/kim# tc qdisc show dev eth0
qdisc netem 8002: root refcnt 2 limit 1000 delay 50.0ms
```

그림 4. interface 설정 변경

100ms로 설정된 eth0의 delay를 50ms로 변경하고 설정 값을 확인하는 화면이다. interface에 설정이 추가되어있는 경우, 설정 값을 변경하고 싶을 때 change를 사용한다.

*Interface의 설정 제거하기

```
root@kim-virtual-machine:/home/kim# tc qdisc del dev eth0 root
root@kim-virtual-machine:/home/kim# tc qdisc show dev eth0
qdisc pfifo_fast 0: root refcnt 2 bands 3 priomap 1 2 2 2 1 2 0 0 1 1 1 1 1 1 1
root@kim-virtual-machine:/home/kim#
```

그림 5. interface 설정 제거

eth0에 설정된 모두 설정 값을 제거하고 설정 값을 확인하는 화면이다.

3. Packet Delay (고정값)

```
root@kim-virtual-machine:/home/kim# tc qdisc change dev eth0 root netem delay 100ms
root@kim-virtual-machine:/home/kim# tc qdisc show dev eth0
qdisc netem 8003: root refcnt 2 limit 1000 delay 100.0ms
root@kim-virtual-machine:/home/kim# ping 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data:
64 bytes from 1.1.1.1: icmp_req=1 ttl=64 time=202 ms
64 bytes from 1.1.1.1: icmp_req=2 ttl=64 time=101 ms
64 bytes from 1.1.1.1: icmp_req=3 ttl=64 time=101 ms
64 bytes from 1.1.1.1: icmp_req=4 ttl=64 time=101 ms
64 bytes from 1.1.1.1: icmp_req=5 ttl=64 time=101 ms
64 bytes from 1.1.1.1: icmp_req=6 ttl=64 time=101 ms
64 bytes from 1.1.1.1: icmp_req=7 ttl=64 time=101 ms
64 bytes from 1.1.1.1: icmp_req=8 ttl=64 time=103 ms
64 bytes from 1.1.1.1: icmp_req=9 ttl=64 time=101 ms
64 bytes from 1.1.1.1: icmp_req=10 ttl=64 time=101 ms
^C
--- 1.1.1.1 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9009ms
rtt min/avg/max/mdev = 101.156/111.553/202.319/30.262 ms
```

그림 6. packet delay 100ms 설정 및 ping test

interface를 통과하는 packet에 일정하게 Delay를 주는 기능이다. dev와 root 사이에 interface name을 작성하고 delay 뒤에 원하는 값을 작성한다. ping test를 통해 packet 교환에 delay가 적용되는 것을 확인할 수 있다.

4. Packet Delay (일정 범위값)

```
root@kim-virtual-machine:/home/kim# tc qdisc change dev eth0 root netem delay 100ms 30ms
root@kim-virtual-machine:/home/kim# tc qdisc show dev eth0
qdisc netem 8003: root refcnt 2 limit 1000 delay 100.0ms 30.0ms
root@kim-virtual-machine:/home/kim# ping 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.
64 bytes from 1.1.1.1: icmp_req=1 ttl=64 time=120 ms
64 bytes from 1.1.1.1: icmp_req=2 ttl=64 time=92.1 ms
64 bytes from 1.1.1.1: icmp_req=3 ttl=64 time=92.4 ms
64 bytes from 1.1.1.1: icmp_req=4 ttl=64 time=90.2 ms
64 bytes from 1.1.1.1: icmp_req=5 ttl=64 time=71.4 ms
64 bytes from 1.1.1.1: icmp_req=6 ttl=64 time=87.2 ms
64 bytes from 1.1.1.1: icmp_req=7 ttl=64 time=101 ms
64 bytes from 1.1.1.1: icmp_req=8 ttl=64 time=90.3 ms
64 bytes from 1.1.1.1: icmp_req=9 ttl=64 time=87.3 ms
64 bytes from 1.1.1.1: icmp_req=10 ttl=64 time=89.3 ms
^C
--- 1.1.1.1 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9015ms
rtt min/avg/max/mdev = 71.451/92.226/120.350/11.698 ms
root@kim-virtual-machine:/home/kim#
```

그림 7. packet delay 100±30ms 설정 및 ping test

고정 Delay 설정과 거의 유사하지만, Delay 값을 일정 범위 안에서 무작위로 주는 기능이
다. delay [기준 delay] [± delay] 로 작성한다. ping test를 통해 70~130ms 사이로 delay가
발생하는 것을 확인할 수 있다.

5. Packet Loss

```
root@kim-virtual-machine:/home/kim# tc qdisc change dev eth0 root netem loss 25%
root@kim-virtual-machine:/home/kim# tc qdisc show dev eth0
qdisc netem 8003: root refcnt 2 limit 1000 loss 25%
root@kim-virtual-machine:/home/kim# ping 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.
64 bytes from 1.1.1.1: icmp_req=1 ttl=64 time=0.499 ms
64 bytes from 1.1.1.1: icmp_req=4 ttl=64 time=0.330 ms
64 bytes from 1.1.1.1: icmp_req=6 ttl=64 time=0.175 ms
64 bytes from 1.1.1.1: icmp_req=7 ttl=64 time=0.427 ms
64 bytes from 1.1.1.1: icmp_req=8 ttl=64 time=0.168 ms
64 bytes from 1.1.1.1: icmp_req=9 ttl=64 time=0.850 ms
64 bytes from 1.1.1.1: icmp_req=10 ttl=64 time=0.232 ms
^C
--- 1.1.1.1 ping statistics ---
10 packets transmitted, 7 received, 30% packet loss, time 9003ms
rtt min/avg/max/mdev = 0.168/0.383/0.850/0.223 ms
```

그림 8. packet loss 25% 설정 및 ping test 1

interface를 통과하는 packet이 일정한 확률로 loss가 발생하도록 만드는 기능이다. loss
후에 %로 loss율을 설정하면 된다. 표본이 많을수록 설정한 값과 유사해지며 <그림 9.> 에
서 확인할 수 있다.

```

64 bytes from 1.1.1.1: icmp_req=13396 ttl=64 time=0.169 ms
64 bytes from 1.1.1.1: icmp_req=13397 ttl=64 time=0.172 ms
64 bytes from 1.1.1.1: icmp_req=13398 ttl=64 time=0.921 ms
64 bytes from 1.1.1.1: icmp_req=13399 ttl=64 time=0.231 ms
64 bytes from 1.1.1.1: icmp_req=13401 ttl=64 time=0.166 ms
64 bytes from 1.1.1.1: icmp_req=13403 ttl=64 time=0.174 ms
64 bytes from 1.1.1.1: icmp_req=13404 ttl=64 time=0.165 ms
64 bytes from 1.1.1.1: icmp_req=13405 ttl=64 time=0.222 ms
64 bytes from 1.1.1.1: icmp_req=13406 ttl=64 time=0.170 ms
64 bytes from 1.1.1.1: icmp_req=13407 ttl=64 time=0.167 ms
64 bytes from 1.1.1.1: icmp_req=13408 ttl=64 time=0.216 ms
^C
--- 1.1.1.1 ping statistics ---
13408 packets transmitted, 10033 received, 25% packet loss, time 13410132ms
rtt min/avg/max/mdev = 0.118/0.434/999.743/9.993 ms, pipe 2
root@kim-virtual-machine:/home/kim#

```

그림 9. packet loss 25% 설정 및 ping test 2

<그림 9>는 약 1만개의 packet을 교환한 테스트이고, 25%의 loss율을 보이고 있음을 확인할 수 있다.

6. Packet Duplication

```

root@kim-virtual-machine:/home/kim# tc qdisc change dev eth0 root netem duplicate 50%
root@kim-virtual-machine:/home/kim# tc qdisc show dev eth0
qdisc netem 8003: root refcnt 2 limit 1000 duplicate 50%
root@kim-virtual-machine:/home/kim# ping 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data:
64 bytes from 1.1.1.1: icmp_req=1 ttl=64 time=0.782 ms
64 bytes from 1.1.1.1: icmp_req=2 ttl=64 time=0.375 ms
64 bytes from 1.1.1.1: icmp_req=2 ttl=64 time=0.448 ms (DUP!)
64 bytes from 1.1.1.1: icmp_req=3 ttl=64 time=0.547 ms
64 bytes from 1.1.1.1: icmp_req=4 ttl=64 time=0.284 ms
64 bytes from 1.1.1.1: icmp_req=5 ttl=64 time=0.376 ms
64 bytes from 1.1.1.1: icmp_req=5 ttl=64 time=0.479 ms (DUP!)
64 bytes from 1.1.1.1: icmp_req=6 ttl=64 time=0.321 ms
64 bytes from 1.1.1.1: icmp_req=7 ttl=64 time=0.326 ms
64 bytes from 1.1.1.1: icmp_req=8 ttl=64 time=0.274 ms
64 bytes from 1.1.1.1: icmp_req=9 ttl=64 time=0.263 ms
64 bytes from 1.1.1.1: icmp_req=10 ttl=64 time=0.425 ms
64 bytes from 1.1.1.1: icmp_req=10 ttl=64 time=0.443 ms (DUP!)
^C
--- 1.1.1.1 ping statistics ---
10 packets transmitted, 10 received, +3 duplicates, 0% packet loss, time 8998ms
rtt min/avg/max/mdev = 0.263/0.411/0.782/0.135 ms

```

그림 10. packet duplication 50% 설정 및 ping test

interface를 통과하는 packet에 일정한 확률로 duplication이 발생하도록 만드는 기능이다. ping test에서는 3개의 duplication의 발생했으나, 표본이 많아지면 설정 값에 가까워진다.

7. Packet Corruption

```
root@kim-virtual-machine:/home/kim# tc qdisc change dev eth0 root netem corrupt 20%
root@kim-virtual-machine:/home/kim# tc qdisc show dev eth0
qdisc netem 8003: root refcnt 2 limit 1000 corrupt 20%
root@kim-virtual-machine:/home/kim# ping 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.
64 bytes from 1.1.1.1: icmp_req=1 ttl=64 time=0.770 ms
64 bytes from 1.1.1.1: icmp_req=2 ttl=64 time=0.325 ms
64 bytes from 1.1.1.1: icmp_req=3 ttl=64 time=0.335 ms
64 bytes from 1.1.1.1: icmp_req=5 ttl=64 time=0.324 ms
64 bytes from 1.1.1.1: icmp_req=6 ttl=64 time=0.515 ms
64 bytes from 1.1.1.1: icmp_req=7 ttl=64 time=0.323 ms
64 bytes from 1.1.1.1: icmp_req=8 ttl=64 time=0.388 ms
64 bytes from 1.1.1.1: icmp_req=9 ttl=64 time=0.330 ms
64 bytes from 1.1.1.1: icmp_req=10 ttl=64 time=0.390 ms
64 bytes from 1.1.1.1: icmp_req=11 ttl=64 time=0.419 ms
64 bytes from 1.1.1.1: icmp_req=12 ttl=64 time=0.323 ms
64 bytes from 1.1.1.1: icmp_req=14 ttl=64 time=0.424 ms
64 bytes from 1.1.1.1: icmp_req=15 ttl=64 time=0.187 ms
^C
--- 1.1.1.1 ping statistics ---
15 packets transmitted, 13 received, 13% packet loss, time 14002ms
rtt min/avg/max/mdev = 0.187/0.388/0.770/0.134 ms
```

그림 11. packet corruption 20% 설정 및 ping test

interface를 통과하는 packet의 offset에 일정 확률로 무작위의 단일 비트 에러를 만드는 기능이다. ping test를 통해서 비트의 에러가 발생하여 데이터 교환이 제대로 이루어지지 않는 것을 확인할 수 있다.

8. 다중 설정

위에 설명한 기능들을 동시에 적용하는 것도 가능하다. parameter들을 뒤에 이어서 작성 해주면 된다.

```
root@kim-virtual-machine:/home/kim# tc qdisc change dev eth0 root netem delay 250ms loss 25%
root@kim-virtual-machine:/home/kim# tc qdisc show dev eth0
qdisc netem 8003: root refcnt 2 limit 1000 delay 250.0ms loss 25%
root@kim-virtual-machine:/home/kim# ping 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data:
64 bytes from 1.1.1.1: icmp_req=1 ttl=64 time=251 ms
64 bytes from 1.1.1.1: icmp_req=5 ttl=64 time=250 ms
64 bytes from 1.1.1.1: icmp_req=6 ttl=64 time=251 ms
64 bytes from 1.1.1.1: icmp_req=7 ttl=64 time=251 ms
64 bytes from 1.1.1.1: icmp_req=8 ttl=64 time=251 ms
64 bytes from 1.1.1.1: icmp_req=10 ttl=64 time=251 ms
64 bytes from 1.1.1.1: icmp_req=11 ttl=64 time=251 ms
64 bytes from 1.1.1.1: icmp_req=12 ttl=64 time=252 ms
64 bytes from 1.1.1.1: icmp_req=13 ttl=64 time=251 ms
64 bytes from 1.1.1.1: icmp_req=15 ttl=64 time=251 ms
^C
--- 1.1.1.1 ping statistics ---
15 packets transmitted, 10 received, 33% packet loss, time 14049ms
rtt min/avg/max/mdev = 250.554/251.428/252.190/0.701 ms
root@kim-virtual-machine:/home/kim#
```

그림 12. packet delay 250ms 와 packet loss 25% 설정 및 ping test

ping test를 통해 packet의 delay가 250ms 와 loss율이 발생하는 것을 확인할 수 있다.

참고 문헌

[1] Linux Foundation, <http://www.linuxfoundation.org/collaborate/workgroups/networking/netem>

[2] Packet Inside, <http://www.packetinside.com/>