RTB-LB Check List

/etc/security/limits.conf

```
* soft nofile 102400
* hard nofile 102400
* soft nproc 102400
* hard nproc 102400
* soft core unlimited
```

sysctl parameter

backlog

```
# 네트워크 장치 별로 커널이 처리하도록 쌓아두는 queue의 크기.
net.core.netdev_max_backlog = 2500

# listen backlog, 즉 listen()으로 바인딩 된 서버 소켓에서 accept()를 기다리는 ESTABLISHED 상태의 소켓 개수에 관련된 커널 파라미터. (즉, connection
completed)을 위한 queue
net.core.somaxconn = 65000

# SYN_RECEIVED 상태의 소켓(즉, connection incompleted)을 위한 queue
net.ipv4.tcp_max_syn_backlog = 20000

# TCP 소켓 송수신 버퍼 크기 조절
# min default max 형식 (단위: byte)
net.ipv4.tcp_rmem = 4096 87380 16777216  # 수신 버퍼
net.ipv4.tcp_wmem = 4096 65536 16777216  # 송신 버퍼
```

Timeout parameters

https://meetup.toast.com/posts/54

```
net.ipv4.tcp_fin_timeout = 20
net.netfilter.nf_conntrack_tcp_timeout_close = 10
net.netfilter.nf_conntrack_tcp_timeout_close_wait = 60
net.netfilter.nf_conntrack_tcp_timeout_established = 432000
net.netfilter.nf_conntrack_tcp_timeout_fin_wait = 120
net.netfilter.nf_conntrack_tcp_timeout_time_wait = 120
```

TIME_WAIT socket buckets

```
# TIME_WAIT 상태 소켓 개수 제한
net.ipv4.tcp_max_tw_buckets = 66024960
```

TCP Fastopen

```

# tcp fastopen(TFO)
# 클라이언트와 서버에서 모두 TCP Fast Open
net.ipv4.tcp_fastopen = 3
# TFO 비활성화 유지 시간 3600초
net.ipv4.tcp_fastopen_blackhole_timeout_sec = 3600
```

packet binding

```

# 로컬 호스트 주소 이외의 다른 가상 IP에 바인딩
net.ipv4.ip_nonlocal_bind = 1

# packet forwarding (네트워크 라우팅 enable)
net.ipv4.ip_forward = 1
```

etc

```
net.ipv4.ip_local_port_range = 1024     65535

# 인터페이스에 할당되지 않은 IP로 소켓 바인딩 - VIP 설정시 필요
net.ipv4.ip_nonlocal_bind = 1
```

HAProxy config

defaults section

```
        option  forwardfor except 127.0.0.0/8
        option  redispatch
```

SSL tune 관련 설정 (haproxy TCP option)

SSL cachesize
이것은 HAProxy 인스턴스가 SSL을 제공하도록 구성된 경우 조정해야 하는 중요한 구성 값입니다.
기본적으로 캐시 크기는 20k이고 캐시는 프로세스 간에 공유됩니다.
각 SSL 연결은 캐시에 항목을 생성합니다.
캐시 크기보다 많은 연결이 있는 경우 가장 오래된 항목이 제거됩니다.
실제로 이것은 이 값보다 더 많은 동시 연결 사용자가 있는 경우 모두 CPU 집약적인 SSL 핸드셰이크를 계속 재발행하여 이 숫자에 도달하면 갑작스러운 성능 저하를 일으킬 수 있음을 의미합니다.
각 항목에 ~200바이트의 메모리가 필요하다는 점을 염두에 두고 이 값을 훨씬 더 높은 숫자로 구성하는 것이 가장 좋습니다.

```
global
    tune.ssl.cachesize 1000000
    (...)
```

값을 100만으로 설정하면 캐시를 보유하기 위해 최소 200MB의 메모리가 필요합니다.

stick-table 적용

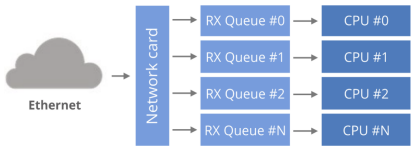
- sticky-session 사용 (세션 동기화)
- spillover 기능 구현
- 참고
  - https://www.haproxy.com/blog/enable-sticky-sessions-in-haproxy/
  - https://www.haproxy.com/blog/introduction-to-haproxy-stick-tables/
  - https://www.haproxy.com/documentation/hapee/latest/configuration/stick-tables/syntax/

5. NIC Inbound / OutBound 설정

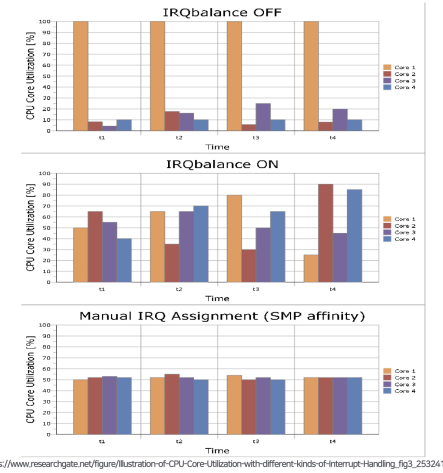
- 한개의 렌카드에서 port0: external / port1: internal
- 따라서, 각 랜카드 port0 bonding / port1 bonding

6. IRQ 지정 참고 URL

2.3 Network IRQ Affinity



https://singhblogging.wordpress.com/2017/02/10/crash-course-to-multi-queue-nics/



s://www.researchgate.net/figure/Illustration-of-CPU-Core-Utilization-with-different-kinds-of-interrupt-Handling_fig3_253241645

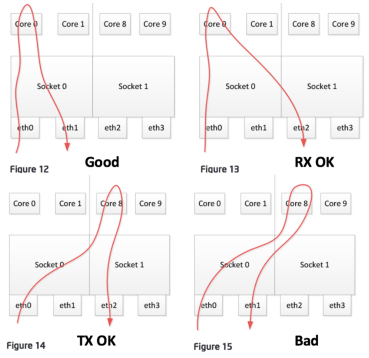


Figure 12 Good Figure 13 RX OK

Figure 14 TX OK Figure 15 Bad

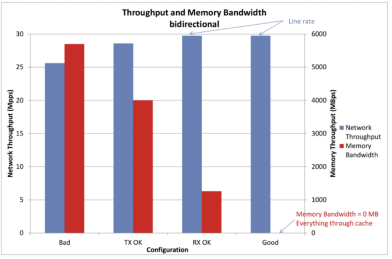


Figure 16

http://docplayer.net/5271505-Network-function-virtualization-virtualized-bras-with-linux-and-intel-

NIC queue에 core 할당

numa node에 맞는 core 할당 필요

Numa node 체크

```
# cat /sys/class/net/eth?/device/local_cpulist
# lstopo  ( hwloc 설치 필요 )
```

Irqbalance stop 및 affinity 설정

```
# systemctl  stop Irqbalance
# set_irq_affinity {core} eth?
Ex ) echo 'core bit' > /proc/irq/$IRQ/smp_affinity
```

Tunnel interface

Single Queue로 Core 분배 어려움

RPS 로 Multi Core 지정 필요

```
# echo 'ffffff' > /sys/class/net/tunl0/queues/rx-0/rps_cpus
```