

# Ttraffic Control для небольшой сети

## Очереди

Существуют несколько видов очередей, соответствующих моделям из теории массового обслуживания и телетрафика:

CBQ (Class-Based Queueing) – Очередь, основанная на классах. Позволяет создавать иерархию классов с приоритетами. Управление производится в зависимости от общей пропускной способности и загруженности канала.

HTB (Hierarchical Token Bucket) – Алгоритм дырявого ведра. Позволяет ограничивать пропускную способность независимо от общей пропускной способности канала. Поддерживает приоритеты.

PRIО – очередь, не влияющая на пропускную способность. Пакеты более низких приоритетов уходят на сетевую карту только при отсутствии в очереди пакетов более высоких приоритетов.

Ещё есть несколько бесклассовых видов очередей:

[P|B]FIFO – First In, First Out с ограничением в пакетах или байтах.

RED (Random Early Detection) – имитирует физические заторы, отбрасывая пакеты при приближении к выделенной полосе пропускания.

SFQ (Stochastic Fairness Queueing) – равномерное распределение пакетов из разных сессий внутри очереди. Трафик разбивается на большое количество FIFO очередей, которые получают доступ к сетевой карте в режиме round-robin. Параметры: perturb – время, через которое будет меняться алгоритм распределения пакетов (обычно ставят 10 с), quantum – объем данных, выпускаемых из потока за один раз (по умолчанию MTU).

TBF (Token Bucket Filter) – замедляет трафик по точно заданному лимиту. Параметры следующие: rate – нужная скорость, latency – максимальное время в очереди, burst – размер “ведра” в байтах.

Если подвести итог, то TBF хорошо подходит, если надо замедлить весь исходящий трафик от узла на конкретном интерфейсе, а HTB – для построения иерархии классов и шейпинга трафика в соответствии с этой иерархией. При этом CBQ требует учитывания ряда дополнительных параметров, которые придётся для каждого конкретного случая получать эмпирическим путём, поэтому используется редко.

\*\*\*\*\*

Определим корневую дисциплину для конкретного сетевого интерфейса и набор классов-наследников:

```
1.tc qdisc add dev eth0 root handle 1: htb default 3
2.tc class add dev eth0 parent 1: classid 1:1 htb rate 64kbit
3.tc class add dev eth0 parent 1: classid 1:2 htb rate 4mbit
burst 15k
4.tc class add dev eth0 parent 1: classid 1:3 htb rate 4mbit
ceil 100mbit
```

Здесь мы создали корневой qdisc с идентификатором 1: , дисциплиной обслуживания HTB, сетевым интерфейсом eth0 и классом по умолчанию 1:3. И для него 3 класса: один с постоянной полосой 64kbit, второй с постоянной полосой 4mbit и максимальным объемом данных, который может быть передан через сетевую карту без переключения на другие классы, 15k, и, наконец, третий с гарантированной скоростью 4mbit и выше.

Для каждого класса при необходимости можно прописать отдельную дисциплину, например:

```
1.tc qdisc add dev eth0 parent 1:1 red min 1600 max 3210 burst
2 limit 32100 avpkt 1000
```

```
2.tc qdisc add dev eth0 parent 1:2 sfq perturb 10
```

После создания иерархии классов и набора дисциплин нам останется только распределить трафик по классам. Сделать это можно несколькими способами.

**Способ первый – tc filter.** В зависимости от некоторых условий направляем пакеты в класс 1:1

```
# tc filter add dev eth0 protocol ip parent 1: prio 1 u32  
match ip dst 192.168.0.1 flowid 1:1
```

В данном случае это пакеты, получателем у которых указан ip 192.168.0.1.

**Способ второй – пометка пакетов с помощью iptables**

```
#iptables -A OUTPUT -t mangle -p tcp -d 192.168.0.2 -dport 80  
-j MARK --set-mark 20
```

```
# tc filter add dev eth0 protocol ip parent 1: prio 2 handle  
20 fw classid 1:2
```

**Способ третий – установка класса сразу с помощью iptables**

```
# iptables -t mangle -A POSTROUTING -p tcp -d 192.168.0.3 -j  
CLASSIFY --set-class 1:3
```

### **Пример 1**

Задачу по ограничению исходящего трафика в направлении определенного хоста можно выполнить, например, следующими тремя командами:

```
#tc qdisc add dev eth0 root handle 1: htb default 1  
#tc class add dev eth0 parent 1: classid 1:1 htb rate 1mbit  
#iptables -t mangle -A POSTROUTING -p tcp -d 192.168.0.3 -j  
CLASSIFY --set-class 1:1
```

Проверить маркировку пакетов в iptables:

```
# iptables -L -n -v -t mangle
```

Удалить все правила:

```
# tc qdisc del dev eth0 root
# iptables -t mangle -F
```

Проверить дисциплину:

```
# tc qdisc
qdisc noqueue 0: dev lo root refcnt 2
qdisc pfifo_fast 0: dev enp0s3 root refcnt 2 bands 3 priomap 1
2 2 2 1 2 0 0 1 1 1 1 1 1 1 1
qdisc pfifo_fast 0: dev enp0s8 root refcnt 2 bands 3 priomap 1
2 2 2 1 2 0 0 1 1 1 1 1 1 1 1
```

Проверить классы:

```
# tc -d -s qdisc show dev enp0s8
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
Sent 1028 bytes 14 pkt (dropped 0, overlimits 0 requeues 0)
backlog 0b 0p requeues 0
```

После выполнения скрипта, картина меняется:

```
# tc qdisc
qdisc noqueue 0: dev lo root refcnt 2
qdisc pfifo_fast 0: dev enp0s3 root refcnt 2 bands 3 priomap 1
2 2 2 1 2 0 0 1 1 1 1 1 1 1 1
qdisc htb 1: dev enp0s8 root refcnt 2 r2q 10 default 6
direct_packets_stat 0 direct_qlen 1000
```

```
# tc -d -s class show dev enp0s8
class htb 1:1 root rate 2Mbit ceil 2Mbit linklayer ethernet
burst 1600b/1 mpu 0b cburst 1600b/1 mpu 0b level 7
Sent 0 bytes 0 pkt (dropped 0, overlimits 0 requeues 0)
backlog 0b 0p requeues 0
lended: 0 borrowed: 0 giants: 0
tokens: 100000 ctokens: 100000
```

```
class htb 1:5 parent 1:1 prio 0 quantum 12500 rate 1Mbit ceil
1500Kbit linklayer ethernet burst 1600b/1 mpu 0b cburst
1599b/1 mpu 0b level 0
Sent 0 bytes 0 pkt (dropped 0, overlimits 0 requeues 0)
backlog 0b 0p requeues 0
lended: 0 borrowed: 0 giants: 0
```

tokens: 200000 ctokens: 133328

```
class htb 1:6 parent 1:1 prio 0 quantum 6250 rate 500Kbit ceil
1500Kbit linklayer ethernet burst 1600b/1 mpu 0b cburst
1599b/1 mpu 0b level 0
```

Sent 0 bytes 0 pkt (dropped 0, overlimits 0 requeues 0)

backlog 0b 0p requeues 0

lended: 0 borrowed: 0 giants: 0

tokens: 400000 ctokens: 133328

```
class htb 1:7 parent 1:1 prio 0 quantum 2500 rate 200Kbit ceil
1Mbit linklayer ethernet burst 1600b/1 mpu 0b cburst 1600b/1
mpu 0b level 0
```

Sent 0 bytes 0 pkt (dropped 0, overlimits 0 requeues 0)

backlog 0b 0p requeues 0

lended: 0 borrowed: 0 giants: 0

tokens: 1000000 ctokens: 200000

Проверить статус:

```
# tc -s -d qdisc show dev eth0
```