tcp 几个新特性介绍

潘卫平
2014.10

Who am I ?

https://git.kernel.org/cgit/linux/kernel/git/torvalds/linux.git/log/?qt=author&q=Weiping

kernel/git/torvalds/linux.git - Linux kernel source tree - Vimperator

kernel git - G... | Kernel.org gi... | The SO_REU... | 163网易免费... | kernel/git/tor...

https://git.kernel.org/cgit/linux/kernel/git/torvalds/linux.git/log/?qt=author&q=Weiping

kernel git

index : kernel/git/torvalds/linux.git

Linux kernel source tree

master switch

Linus Torvalds

summary refs log tree commit diff stats

author Weiping search

Age	Commit message (Expand)	Author	Files	Lines
2014-09-28	tcp: use tcp_flags in tcp_data_queue()	Peter Pan(潘卫平)	1	-3/+2
2014-06-17	net: delete duplicate dev_set_rx_mode() call	Peter Pan(潘卫平)	1	-5/+1
2014-04-20	tcp: make tcp_cwnd_application_limited() static	Weiping Pan	3	-23/+22
2014-03-26	tcp: delete unused parameter in tcp_nagle_check()	Peter Pan(潘卫平)	1	-3/+3
2014-01-22	tcp: delete redundant calls of tcp_mtup_init()	Peter Pan(潘卫平)	2	-2/+0
2013-09-30	Documentation/kernel-parameters.txt: replace kernelcore with Movable	Weiping Pan	1	-1/+1
2013-08-14	tun: compare with 0 instead of total_len	Weiping Pan	1	-2/+2
2013-06-20	tcp:typo unset should be unsent	Weiping Pan	1	-1/+1
2013-06-11	net: pass correct parameter to skb_headers_offset_update()	Peter Pan(潘卫平)	1	-13/+2
2012-10-01	use skb_end_offset() in skb_try_coalesce()	Weiping Pan	1	-2/+1
2012-07-23	rds: set correct msg_namelen	Weiping Pan	1	-0/+3
2012-06-12	bonding:record primary when modify it via sysfs	Weiping Pan	1	-2/+6
2012-03-29	bonding: emit event when bonding changes MAC	Weiping Pan	1	-1/+7
2012-03-19	bonding: send igmp report for its master	Peter Pan(潘卫平)	1	-3/+15
2011-12-06	ipv4:correct description for tcp_max_syn_backlog	Peter Pan(潘卫平)	2	-8/+9
2011-12-02	ipv4: flush route cache after change accept_local	Peter Pan(潘卫平)	1	-0/+5
2011-11-01	bonding:update speed/duplex for NETDEV_CHANGE	Weiping Pan	2	-27/+22
2011-08-18	bonding:reset backup and inactive flag of slave	Peter Pan(潘卫平)	1	-0/+18
2011-06-13	bonding:delete a dereference before check	Peter Pan(潘卫平)	1	-2/+1
2011-06-09	bonding: delete unused arp_mon_pt	Peter Pan(潘卫平)	1	-1/+0
2011-06-09	bonding: delete unused ad_timer	Peter Pan(潘卫平)	1	-1/+0
2011-06-09	bonding:delete agg_select_mode from ad_bond_info	Peter Pan(潘卫平)	2	-3/+1
2011-06-09	bonding:delete lacp_fast from ad_bond_info	Peter Pan(潘卫平)	3	-10/+5
2011-06-09	bonding: make 802.3ad use latest lacp_rate	Peter Pan(潘卫平)	3	-0/+33
2011-05-27	platform/x86:delete two unused variables	Weiping Pan	1	-2/+0
2011-05-13	net:set valid name before calling ndo_init()	Peter Pan(潘卫平)	1	-4/+4
2011-04-26	staging: comedi: fix a typo	Weiping Pan(潘卫平)	1	-1/+1
2011-04-13	usb: fix a typo in a comment	Weiping Pan(潘卫平)	1	-1/+1
2011-04-11	bonding:fix two typos	Peter Pan(潘卫平)	1	-2/+2
2011-04-11	net: fix tranmitted/tranmitting typo	Peter Pan(潘卫平)	4	-5/+5
2011-04-11	bonding:delete unused rlb_interval_counter	Peter Pan(潘卫平)	1	-1/+0
2011-04-11	bonding:delete unused alb_timer	Peter Pan(潘卫平)	1	-1/+0
2011-04-11	bonding:set save_load to 0 when initializing	Peter Pan(潘卫平)	1	-1/+1
2011-03-31	bonding:typo in comment	Peter Pan(潘卫平)	1	-1/+1

generated by cgit v0.10.1 at 2014-10-28 10:05:23 (GMT)

Who am I ?

tcp : bash – Konsole

File Edit View Bookmarks Settings Help

```
[pwp@localhost tcp]$
[pwp@localhost tcp]$
[pwp@localhost tcp]$ ls
1063-1191       2883       4737              fast_open       rtt
1072-1185-1323  2988-6298  5236              my_doc          spurious_fast_retransmission_eifel
1072-2018       3042       5482              nagle-minshall  spurious_retransmission_timeout_frto
1112            3465       5827              paper           tcp_options
1122            3517-6675  5961              patches         tcp_state_machine
2001-2581-5681  3522       793               prr             tcp_todo
2385-5925       3708       896               rate_halving    tlp
2582-3782-6582  4015       early_retransmit  reordering
2861            4138-5682  fack              rhel_books
[pwp@localhost tcp]$
[pwp@localhost tcp]$
[pwp@localhost tcp]$
[pwp@localhost tcp]$
[pwp@localhost tcp]$
[pwp@localhost tcp]$
[pwp@localhost tcp]$
[pwp@localhost tcp]$
[pwp@localhost tcp]$
[pwp@localhost tcp]$
[pwp@localhost tcp]$
```

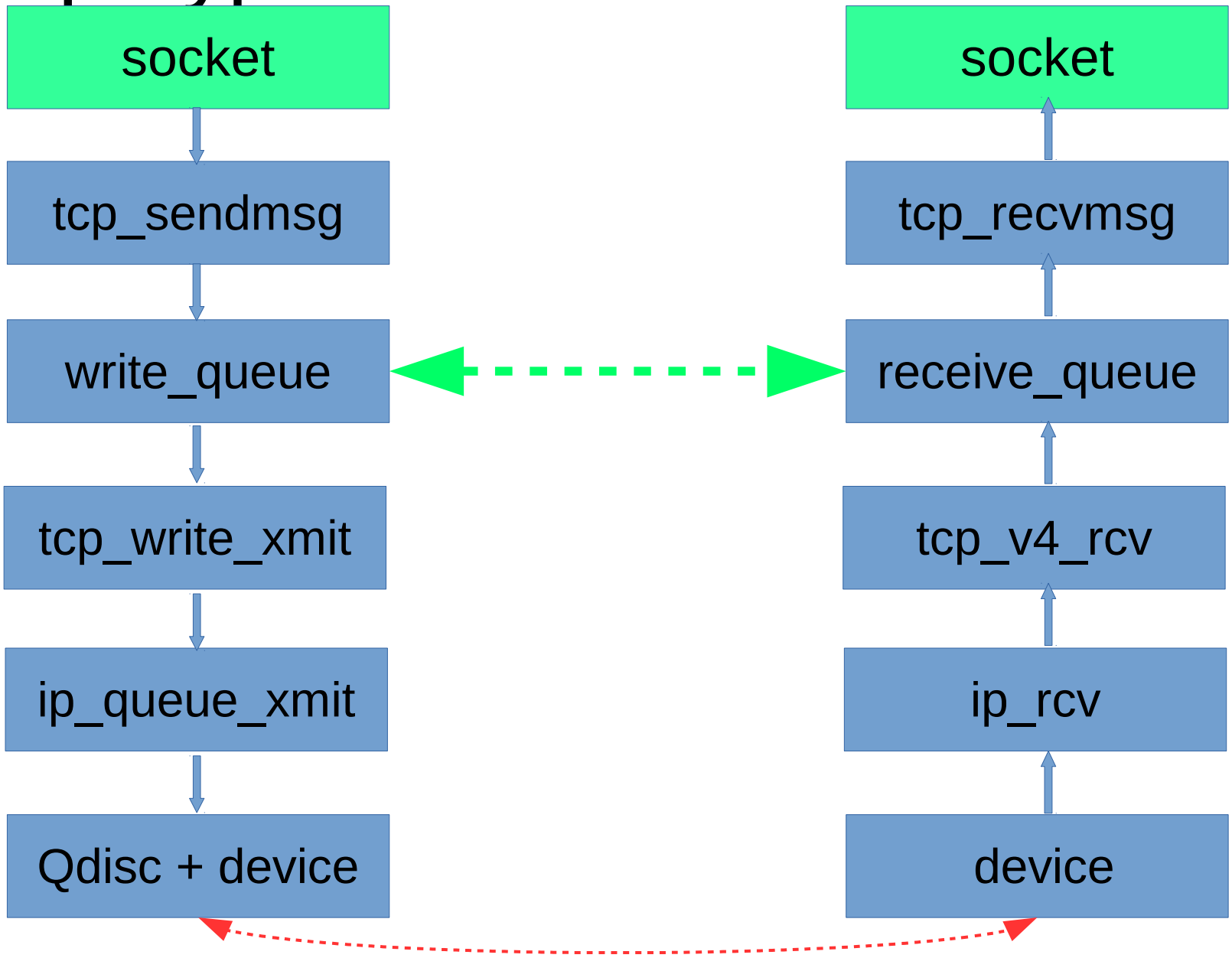
41-socket-reuseport : bash | 41-socket-reuseport : bash | net : git | net : bash | net : bash | my_doc : oosplash | prr : bash | tcp : bash

The SO_REUSEPORT socket opti | tcp : bash – Konsole | tcp_new_features.odp - LibreOffi | 06:01 P | US

提纲—— tcp 几个新特性

1. tcp bypass
2. ER early retransmit
3. TLP tail loss probe
4. PRR proportional rate reduction
5. TSQ tcp small queues
6. SO_REUSEPORT

1. tcp bypass


socket
tcp_sendmsg
write_queue
tcp_write_xmit
ip_queue_xmit
Qdisc + device
socket
tcp_recvmsg
receive_queue
tcp_v4_rcv
ip_rcv
device

MS	BASE	AF_UNIX	BYPASS		TCP_STREAM_MS
1	15.64	5.90	5.12	32%	86%
2	30.93	9.81	10.48	33%	106%
4	58.22	19.70	21.29	36%	108%
8	117.00	39.00	42.74	36%	109%
16	231.08	84.59	83.90	36%	99%
32	439.39	159.93	163.03	37%	101%
64	879.13	323.31	322.78	36%	99%
128	1617.55	632.50	646.34	39%	102%
256	3091.72	1316.36	1206.93	39%	91%
512	5077.18	2359.51	2342.00	46%	99%
1024	7403.20	6302.20	3335.23	45%	52%
2048	10194.40	13922.19	5751.23	56%	41%
4096	13338.08	22566.45	9447.29	70%	41%
8192	14467.93	28122.20	13758.43	95%	48%
16384	22463.15	37522.42	26804.36	119%	71%
32768	14743.58	30591.61	17040.15	115%	55%
65536	24743.77	33855.93	40418.15	163%	119%
131072	13925.14	31762.52	48292.60	346%	152%
262144	16126.15	32912.89	25610.47	158%	77%
524288	12080.51	35059.27	30608.31	253%	87%
1048576	10539.06	28200.14	16953.69	160%	60%

2.ER —— early retransmit

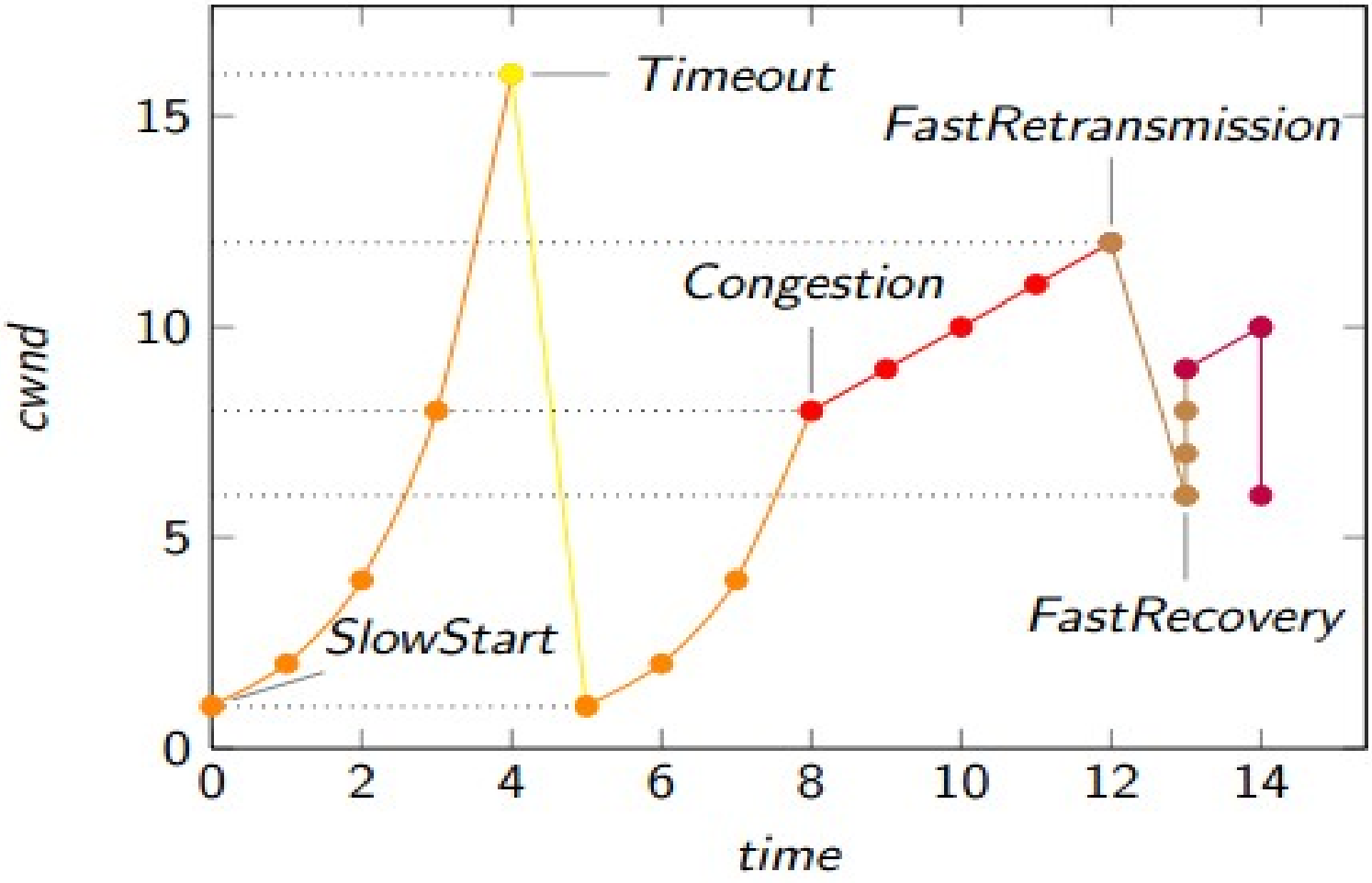
```
static int tcp_time_to_recover(struct sock *sk)
{
        if (tp->do_early_retrans
            && !tp->retrans_out
            && tp->sacked_out
            && (tp->packets_out == (tp->sacked_out + 1)
                && tp->packets_out < 4)
            && !tcp_may_send_now(sk))
            return 1;
}
```

3.TLP——tail loss probe

定时器 max(2*SRTT, 10ms)

```
static bool tcp_write_xmit( )
{
    if (tcp_transmit_skb () == 0)  //  发送成功
        tcp_schedule_loss_probe()
}
```

4.PRR——proportional rate reduction



rfc2001->rfc2581->rfc5681 (without sack)

- 调整　ssthresh = max (cwnd / 2, 2)

  cwnd = ssthresh + 3

  重发 snd_una 包
- 接收到新的重复 ACK　　cwnd = cwnd + 1
- 退出 cwnd = ssthresh（平衡点）

```
static inline unsigned int tcp_packets_in_flight(struct tcp_sock *tp)
{
        return tp->packets_out - tcp_left_out(tp) + tp->retrans_out;
}

static inline unsigned int tcp_left_out(const struct tcp_sock *tp)
{
        return tp->sacked_out + tp->lost_out;
}
```

rfc3517->rfc6675 (with sack)

```
static void tcp_cwnd_down(struct sock *sk)
{
        tp->snd_cwnd = min(tp->snd_cwnd,
                tcp_packets_in_flight(tp) + 1);
}
```

prr 核心算法

```
static void tcp_cwnd_reduction(struct sock *sk, const int
prior_unsacked)
{
        int newly_acked_sacked = prior_unsacked -
                                (tp->packets_out - tp->sacked_out);
        tp->prr_delivered += newly_acked_sacked;
        if (tcp_packets_in_flight(tp) > tp->snd_ssthresh) {
                u64 dividend = (u64)tp->snd_ssthresh * tp->prr_delivered
                                +  tp->prior_cwnd - 1;
                sndcnt = div_u64(dividend, tp->prior_cwnd) - tp->prr_out;
        }
        tp->snd_cwnd = tcp_packets_in_flight(tp) + sndcnt;
}
```

5.TSQ——tcp small queues

- 根据——对速度的估计
- 限制——进入 Qdisc 或驱动的包的数量为 2，或者 为 1ms 时间内能传输的包的个数

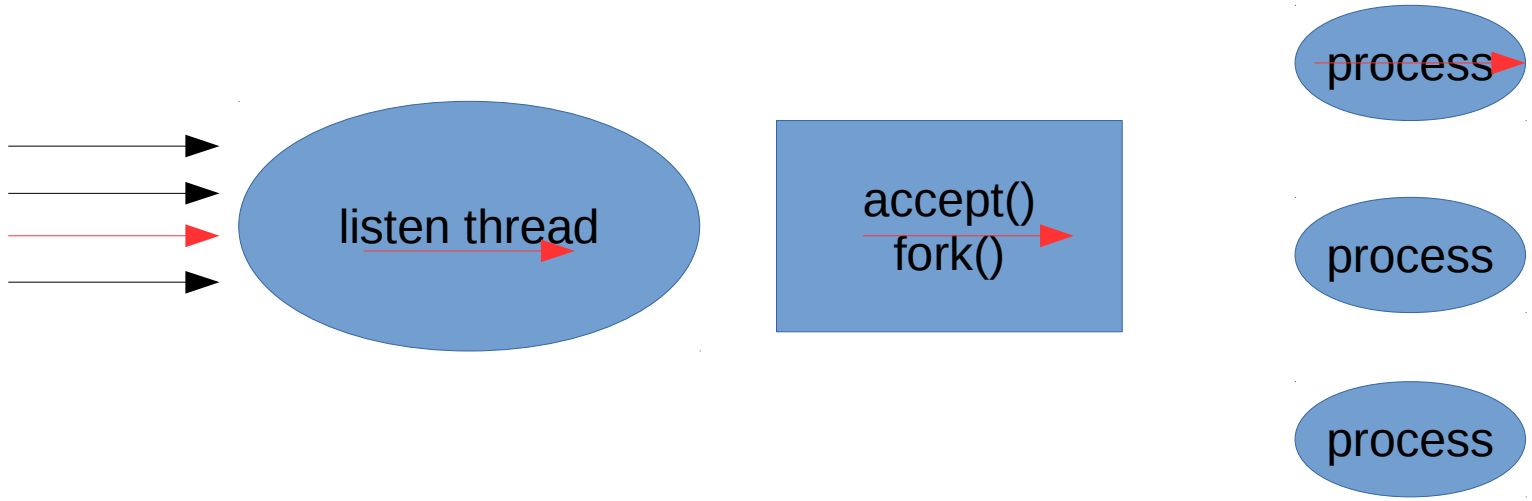
socket 限速 ？

SO_MAX_PACING_RATE

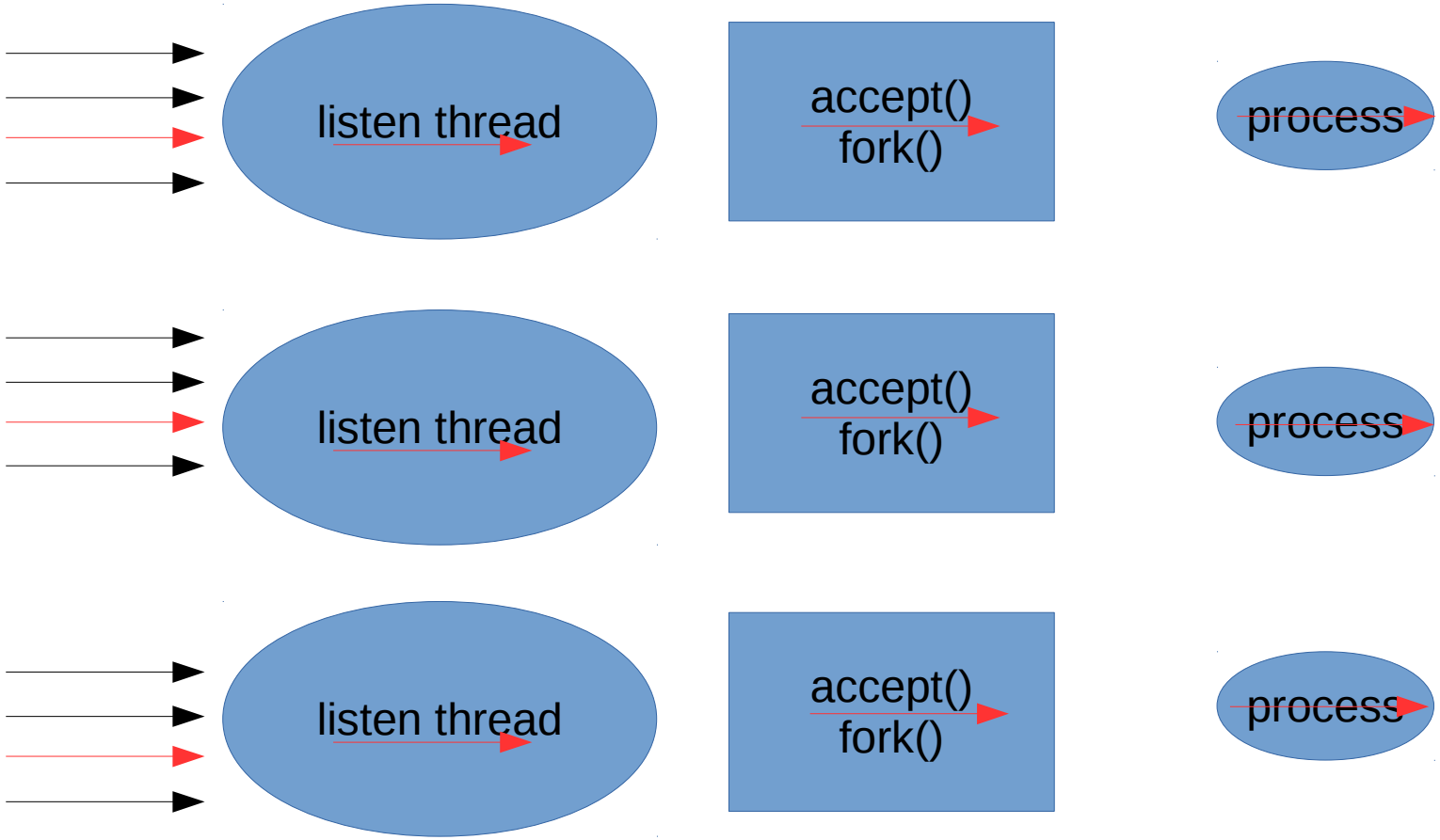
Qdisc fq

6.SO_REUSEPORT

未使用 SO_REUSEPORT



使用 SO_REUSEPORT



What can I do ?