Real world tips, tricks, and notes of using epoll-based busy polling to reduce latency

Joe Damato

Hi, my name is Joe.

I work at Fastly.

My opinions are my own.

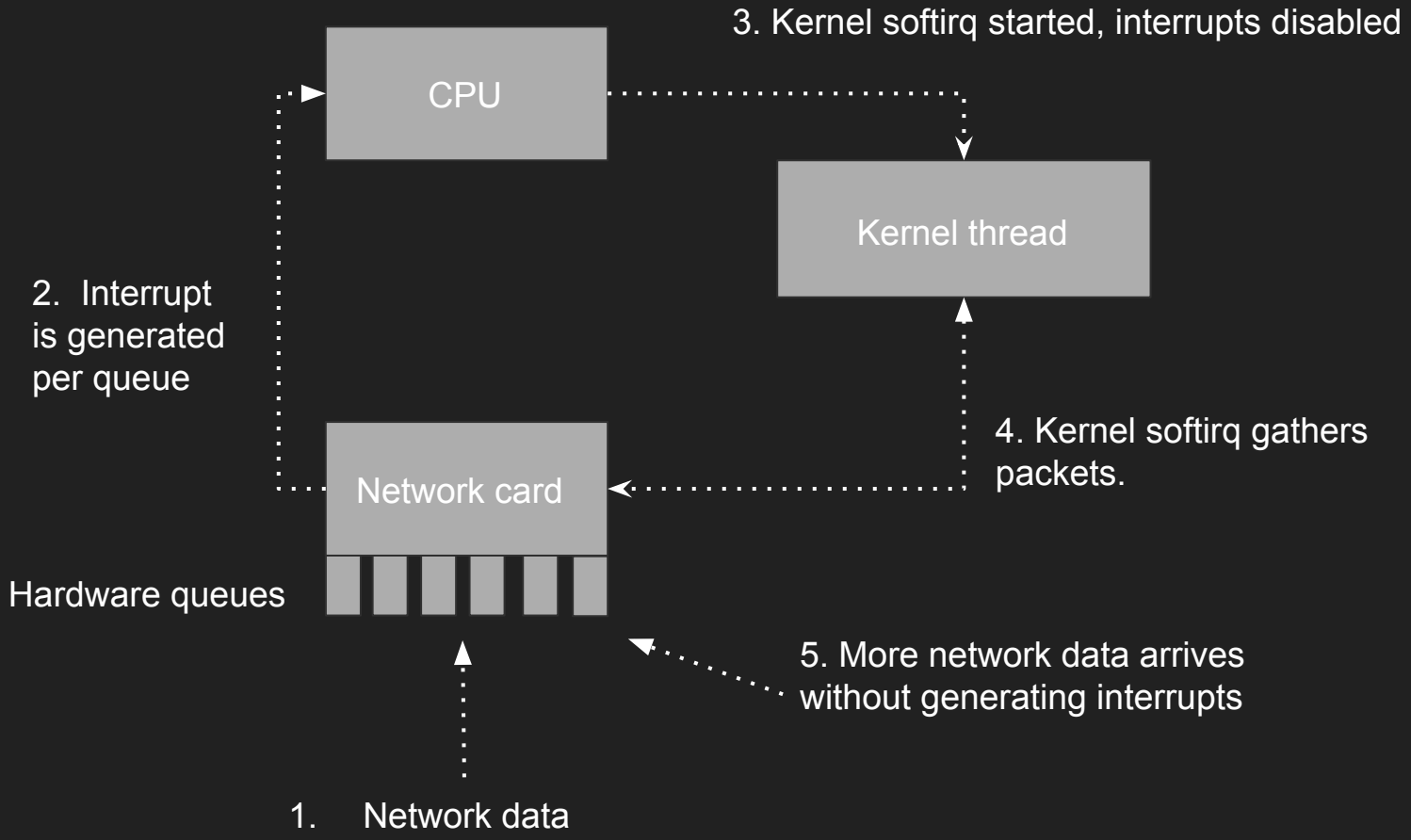
A Fastly computer looks like this

Overall takeaways:

- Busy poll helped us reduce latency in prod
- Needed some tweaks to get us there
- System-wide config is too coarse-grained
- More work to be done
- More docs needed overall

But before we can talk about busy polling: NAPI.

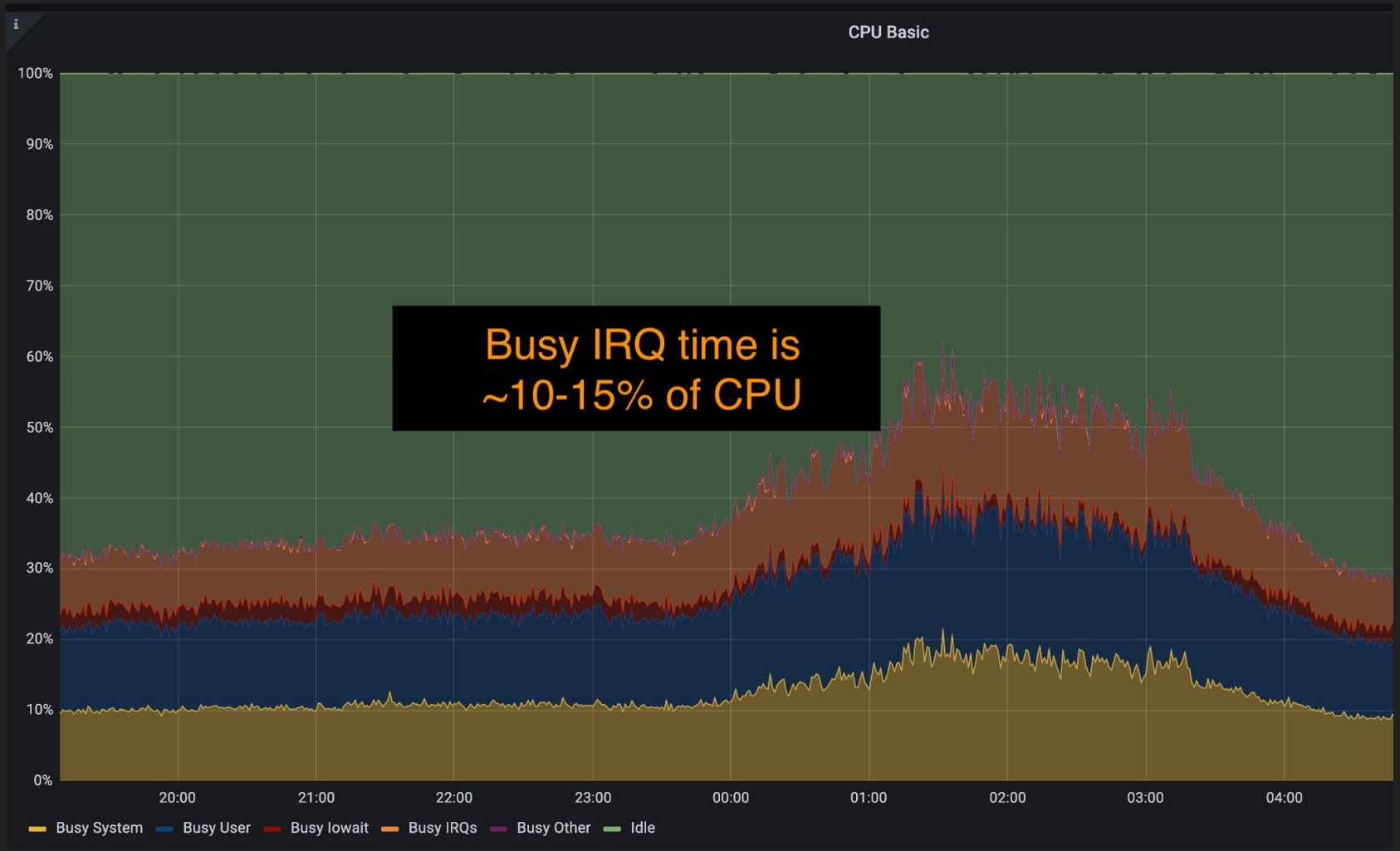
[Very high-level diagram]


3. Kernel softirq started, interrupts disabled
CPU
Kernel thread
2. Interrupt is generated per queue
4. Kernel softirq gathers packets.
Network card
Hardware queues
5. More network data arrives without generating interrupts
1. Network data

NAPI

- Reduce interrupt load
- Improve CPU cache efficiency
- Should benefit high speed NICs (faster speed = faster data arrival = more interrupts)

But … real world data from a prod machine with lots of traffic, suggests….


CPU Basic
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
Busy IRQ time is
~10-15% of CPU
20:00
21:00
22:00
23:00
00:00
01:00
02:00
03:00
04:00
Busy System
Busy User
Busy Iowait
Busy IRQs
Busy Other
Idle

This felt like something was wrong.

So much IRQ traffic.

NAPI efficiency - RX packets per IRQ

First thing I tried, reducing the number of queues and configuring IRQ coalescing.

Simplest thing first: push tx-usecs out

i40e NIC
25gbps NIC
Driver modified to count IRQs
iperf3 test

Zero work readings on i40e during production test

10 second timeout

	test-node-1(reduced q + TX coalesce)	test-node-2 (tx coalesce)	control
Run 1:	4,013,254	4,659,321	4,558,964
Run 2:	4,059,272	4,774,618	4,833,400
Run 3:	4,043,220	4,858,869	4,841,198
Run 4:	4,124,204	4,781,532	4,947,740
Run 5:	3,730,169	4,199,293	4,184,878

60 second time out

	test-node-1 (reduced q + TX coalesce)	test-node-2 (tx-coalesce)	control
Run 1:	25,634,210	30,087,788	30,231,770
Run 2:	25,340,944	29,948,818	30,862,257
Run 3:	23,767,422	27,684,464	28,243,535
Run 4:	22,551,608	26,401,489	26,392,595
Run 5:	22,625,945	26,109,389	26,695,311

TLDR; reduction in “no-work” IRQs with i40e, maybe ~15% or so?

```
        test              control
Run 1: 7,562,960   vs.   7,849,544    (+286,584 no work IRQs on control)
Run 2: 7,461,928   vs.   7,917,925    (+455,997 no work IRQs on control)
Run 3: 7,477,857   vs.   7,546,702    (+68,845 no work IRQs control)
Run 4: 7,456,894   vs.   7,729,559    (+272,665 no work IRQs on control)


Longer test, timeout 60s:

       test            control
Run 1: 47,009,259    vs.   49,755,821   (+2,746,562 no work IRQs on control)
Run 2: 46,678,501        vs.   49,547,378   (+2,868,877 no work IRQs on
control)
Run 3: 46,313,645        vs.   49,955,651   (+3,642,006 no work IRQs on
control)
Run 4: 43,709,998        vs.   47,735,505   (+4,025,504 no work IRQs on control)
```

TLDR; 3-5% improvement in “no-work” IRQS on mlx5

Tiny improvement, but still felt like there was something I was missing?

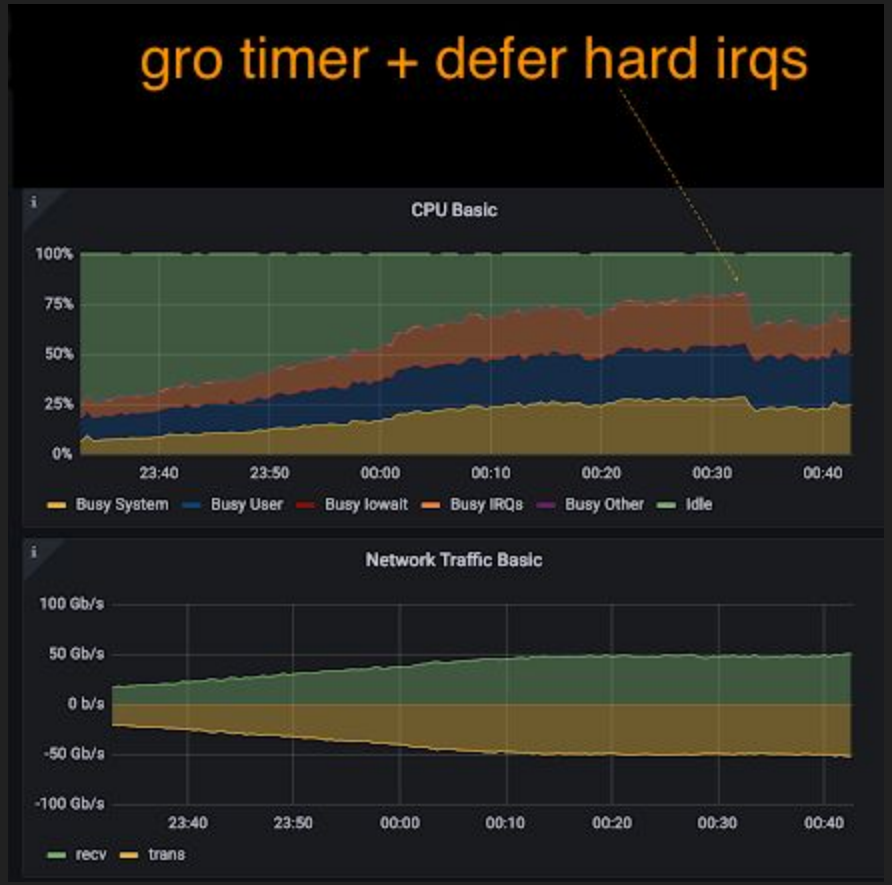
Lots of reading of kernel code

Stumbled on /sys/class/net/*/….

napi_defer_hard_irqs
gro_flush_timeout

These settings are a (system-wide) way to keep NIC IRQs disabled (sw IRQ coalescing)

And process incoming data via a timer instead



Take away:

Controlling RX IRQs reduces busy IRQ tremendously under load

Using these settings reduced CPU usage *considerably*.

… which shows that RX IRQs are a significant source of CPU use.

BUT

Latency was introduced.

Timer fires more slowly than the device fires IRQs.

Lots of reading of kernel code

SO_INCOMING_NAPI_ID

No clue what this thing was or what it did.

Stumbled on an email from Cong Wang from 2019 who also didn’t know what it did

From: Cong Wang 2019-02-14 20:15 UTC (permalink / raw)
  To: Alexander Duyck
  Cc: Eric Dumazet, sridhar.samudrala, Linux Kernel Network Developers

Hello,

While looking into the busy polling in Linux kernel, three questions come into my mind:

1. In the document[1], it claims sysctl.net.busy_poll depends on either SO_BUSY_POLL or sysctl.net.busy_read. However, from the code in ep_set_busy_poll_napi_id(), I don't see such a dependency. It simply checks sysctl_net_busy_poll and sk->sk_napi_id, but sk->sk_napi_id is always set as long as we enable CONFIG_NET_RX_BUSY_POLL. So what I am missing here?

2. Why there is no socket option for sysctl.net.busy_poll? Clearly sysctl_net_busy_poll is global and SO_BUSY_POLL only works for sysctl.net.busy_read.

3. How is SO_INCOMING_NAPI_ID supposed to be used? I can't find any useful documents online. Any example or more detailed doc?

Thanks!

Turns out that Cong Wang writes kernel code

index : kernel/git/netdev/net-next.git

Netdev Group's -next networking tree

main switch

Netdev Group

about summary refs log tree commit diff stats

log msg Cong Wang search

Age	Commit message (Expand)	Author	Files	Lines
2023-07-14	pie: fix kernel-doc notation warning	Randy Dunlap	1	-1/+1
2023-07-14	codel: fix kernel-doc notation warnings	Randy Dunlap	1	-2/+2
2023-06-22	sch_netem: acquire qdisc lock in netem_change()	Eric Dumazet	1	-2/+6
2023-06-09	kexec: enable kexec_crash_size to support two crash kernel regions	Zhen Lei	1	-5/+38
2023-06-09	kexec: add helper __crash_shrink_memory()	Zhen Lei	1	-22/+29
2023-06-09	kexec: improve the readability of crash_shrink_memory()	Zhen Lei	1	-10/+5
2023-06-09	kexec: clear crashk_res if all its memory has been released	Zhen Lei	1	-4/+7
2023-06-09	kexec: delete a useless check in crash_shrink_memory()	Zhen Lei	1	-1/+1
2023-06-09	kexec: fix a memory leak in crash_shrink_memory()	Zhen Lei	1	-3/+2
2023-06-08	kcm: Use splice_eof() to flush	David Howells	1	-0/+15
2023-06-05	kcm: Convert kcm_sendpage() to use MSG_SPLICE_PAGES	David Howells	1	-143/+18
2023-06-05	kcm: Support MSG_SPLICE_PAGES	David Howells	1	-17/+40
2023-04-28	sit: update dev->needed_headroom in ipip6_tunnel_bind_dev()	Cong Wang	1	-3/+5
2023-04-08	notifiers: add tracepoints to the notifiers infrastructure	Guilherme G. Piccoli	2	-0/+75
2023-03-31	virtio/vsock: fix leaks due to missing skb owner	Bobby Eshleman	1	-0/+10
2023-02-20	l2tp: Avoid possible recursive deadlock in l2tp_tunnel_register()	Shigeru Yoshida	1	-58/+67
2023-02-08	bpf/docs: Update design QA to be consistent with kfunc lifecycle docs	Toke Høiland-Jørgensen	1	-7/+18
2023-01-31	net: sched: sch: Bounds check priority	Kees Cook	1	-1/+4
2023-01-25	net: Kconfig: fix spellos	Randy Dunlap	2	-2/+2
2023-01-23	net/sock: Introduce trace_sk_data_ready()	Peilin Ye	31	-0/+128
2023-01-18	l2tp: prevent lockdep issue in l2tp_tunnel_register()	Eric Dumazet	1	-3/+2
2023-01-16	Merge branch 'l2tp-races'	David S. Miller	1	-53/+52
2023-01-16	l2tp: close all race conditions in l2tp_tunnel_register()	Cong Wang	1	-14/+14
2023-01-16	l2tp: convert l2tp_tunnel_list to idr	Cong Wang	1	-43/+42
2023-01-02	net: sched: htb: fix htb_classify() kernel-doc	Randy Dunlap	1	-2/+6
2022-12-26	net: sched: fix memory leak in tcindex_set_parms	Hawkins Jiawei	1	-10/+2
2022-12-19	net_sched: reject TCF_EM_SIMPLE case for complex ematch module	Cong Wang	1	-0/+2
2022-11-15	kcm: close race conditions on sk_receive_queue	Cong Wang	1	-52/+6
2022-11-03	Merge tag 'for-netdev' of https://git.kernel.org/pub/scm/linux/kernel/git/bpf/bpf...	Jakub Kicinski	10	-16/+113
2022-11-03	bpf, sock_map: Move cancel_work_sync() out of sock lock	Cong Wang	3	-9/+7
2022-10-16	skmsg: pass gfp argument to alloc_sk_msg()	Eric Dumazet	1	-4/+4
2022-09-29	net: sched: cls_u32: Avoid memcpy() false-positive warning	Kees Cook	1	-1/+5
2022-09-26	af_unix: Refactor unix_read_skb()	Peilin Ye	1	-24/+10
2022-09-26	udp: Refactor udp_read_skb()	Peilin Ye	1	-29/+17
2022-09-20	tcp: read multiple skbs in tcp_read_skb()	Cong Wang	1	-10/+19
2022-09-09	Merge branch 'tc_action_ops-refactor'	David S. Miller	22	-513/+118
2022-09-08	sch_sfb: Also store skb len before calling child enqueue	Toke Høiland-Jørgensen	1	-1/+2
2022-08-31	kcm: fix strp_init() order and cleanup	Cong Wang	1	-8/+7
2022-08-30	net: sched: using TCQ_MIN_PRIO_BANDS in prio_tune()	Zhengchao Shao	1	-1/+1
2022-08-18	Merge branch 'tcp-some-bug-fixes-for-tcp_read_skb'	Jakub Kicinski	2	-28/+26
2022-08-18	tcp: handle pure FIN case correctly	Cong Wang	2	-3/+4
2022-08-18	tcp: refactor tcp_read_skb() a bit	Cong Wang	1	-17/+9
2022-08-18	tcp: fix tcp_cleanup_rbuf() for tcp_read_skb()	Cong Wang	1	-10/+14
2022-08-18	tcp: fix sock skb accounting in tcp_read_skb()	Cong Wang	1	-0/+1
2022-07-09	Merge https://git.kernel.org/pub/scm/linux/kernel/git/bpf/bpf-next	Jakub Kicinski	125	-6701/+5141
2022-06-20	skmsg: Get rid of unncessary memset()	Cong Wang	1	-10/+13
2022-06-20	skmsg: Get rid of skb_clone()	Cong Wang	2	-6/+2
2022-06-20	net: Introduce a new proto_ops ->read_skb()	Cong Wang	9	-47/+31
2022-06-20	tcp: Introduce tcp_read_skb()	Cong Wang	2	-0/+49
2022-06-17	net/sched: sch_netem: Fix arithmetic in netem_dump() for 32-bit platforms	Peilin Ye	1	-2/+2
2022-05-16	af_unix: Silence randstruct GCC plugin warning	Kees Cook	2	-7/+3
2022-04-19	net: sched: support hash selecting tx queue	Tonghao Zhang	3	-2/+50
2022-04-19	net: sched: use queue_mapping to pick tx queue	Tonghao Zhang	4	-3/+38
2022-04-15	net/sched: cls_u32: fix possible leak in u32_init_knode()	Eric Dumazet	1	-4/+4
2022-04-15	net/sched: cls_u32: fix netns refcount changes in u32_change()	Eric Dumazet	1	-6/+10
2022-04-15	ip6_gre: Fix skb_under_panic in __gre6_xmit()	Peilin Ye	1	-3/+6
2022-04-15	ip6_gre: Avoid updating tunnel->tun_hlen in __gre6_xmit()	Peilin Ye	1	-2/+3
2022-03-15	bpf, sockmap: Fix memleak in tcp_bpf_sendmsg while sk msg is full	Wang Yufen	1	-4/+13
2022-03-03	ipv6: fix skb drops in igmp6_event_query() and igmp6_event_report()	Eric Dumazet	2	-22/+14
2022-02-16	net: sched: limit TC_ACT_REPEAT loops	Eric Dumazet	1	-3/+10
2022-02-14	net_sched: add __rcu annotation to netdev->qdisc	Eric Dumazet	5	-29/+36
2022-02-11	block: introduce block_rq_error tracepoint	Yang Shi	2	-14/+39
2022-02-11	net: dsa: remove lockdep class for DSA slave address list	Vladimir Oltean	1	-12/+0
2022-02-11	net: dsa: remove lockdep class for DSA master address list	Vladimir Oltean	1	-4/+0
2022-02-08	ipmr,ip6mr: acquire RTNL before calling ip[6]mr_free_table() on failure path	Eric Dumazet	2	-0/+4
2022-02-01	net: sched: fix use-after-free in tc_new_tfilter()	Eric Dumazet	1	-3/+37
2021-12-29	net: fix use-after-free in tw_timer_handler	Muchun Song	1	-6/+4
2021-12-09	xfrm: use net device refcount tracker helpers	Eric Dumazet	2	-3/+3
2021-12-02	selftests/fib_tests: Rework fib_rp_filter_test()	Peilin Ye	1	-10/+49
2021-11-26	net/sched: sch_ets: don't peek at classes beyond 'nbands'	Davide Caratti	1	-3/+5
2021-11-23	tc-testing: Add link for reviews with TC MAINTAINERS	Jamal Hadi Salim	1	-0/+1
2021-11-16	selftests: add a test case for mirred egress to ingress	Davide Caratti	2	-1/+47
2021-11-16	net: sched: act_mirred: drop dst for the direction from egress to ingress	Xin Long	1	-3/+8
2021-11-16	Merge https://git.kernel.org/pub/scm/linux/kernel/git/bpf/bpf	Jakub Kicinski	23	-73/+573
2021-11-16	udp: Validate checksum in udp_read_sock()	Cong Wang	1	-0/+11
2021-11-01	skmsg: Lose offset info in sk_psock_skb_ingress	Liu Jian	2	-12/+49
2021-10-26	Merge https://git.kernel.org/pub/scm/linux/kernel/git/bpf/bpf	Jakub Kicinski	23	-98/+118
2021-10-26	Merge branch 'sock_map: fix ->poll() and update selftests'	Alexei Starovoitov	13	-78/+58
2021-10-26	selftests/bpf: Use recv_timeout() instead of retries	Yucong Sun	1	-55/+20
2021-10-26	net: Implement ->sock_is_readable() for UDP and AF_UNIX	Cong Wang	4	-0/+10
2021-10-26	skmsg: Extract and reuse sk_msg_is_readable()	Cong Wang	3	-14/+16
2021-10-26	net: Rename ->stream_memory_read to ->sock_is_readable	Cong Wang	6	-11/+14
2021-10-12	af_unix: Rename UNIX-DGRAM to UNIX to maintain backwards compatability	Stephen Boyd	1	-1/+1
2021-10-08	net/sched: sch_ets: properly init all active DRR list handles	Davide Caratti	1	-3/+9
2021-10-06	unix: Fix an issue in unix_shutdown causing the other end read/write failures	Jiang Wang	1	-4/+5
2021-09-30	net: sched: flower: protect fl_walk() with rcu	Vlad Buslov	1	-0/+6
2021-09-19	net: sched: move and reuse mq_change_real_num_tx()	Jakub Kicinski	4	-46/+27
2021-08-31	af_unix: fix potential NULL deref in unix_dgram_connect()	Eric Dumazet	1	-3/+6
2021-08-30	net: sched: Fix qdisc_rate_table refcount leak when get tcf_block failed	Xiyu Yang	1	-1/+1
2021-08-18	ovs: clear skb->tstamp in forwarding path	kaixi.fan	1	-0/+1
2021-08-16	selftest/bpf: Add new tests in sockmap for unix stream to tcp.	Jiang Wang	1	-4/+12
2021-08-16	selftest/bpf: Change udp to inet in some function names	Jiang Wang	1	-13/+17
2021-08-16	selftest/bpf: Add tests for sockmap with unix stream type.	Jiang Wang	1	-0/+2
2021-08-16	af_unix: Add unix_stream_proto for sockmap	Jiang Wang	4	-37/+148
2021-08-16	af_unix: Add read_sock for stream socket types	Jiang Wang	1	-0/+12
2021-08-10	Merge https://git.kernel.org/pub/scm/linux/kernel/git/bpf/bpf-next	Jakub Kicinski	28	-306/+3431
2021-08-04	docs: networking: netdevsim rules	Jakub Kicinski	1	-0/+17
2021-08-04	tc-testing: Add control-plane selftests for sch_mq	Peilin Ye	2	-0/+138
2021-08-03	bpf, unix: Check socket type in unix_bpf_update_proto()	Cong Wang	1	-0/+3
2021-08-02	net_sched: refactor TC action init API	Cong Wang	37	-169/+185
2021-07-30	unix_bpf: Fix a potential deadlock in unix_dgram_bpf_recvmsg()	Cong Wang	1	-8/+8
2021-07-28	tc-testing: Add control-plane selftest for skbmod SKBMOD_F_ECN option	Peilin Ye	1	-0/+24
2021-07-28	net/sched: act_skbmod: Add SKBMOD_F_ECN option support	Peilin Ye	2	-12/+33
2021-07-27	net: split out SIOCDEVPRIVATE handling from dev_ioctl	Arnd Bergmann	3	-3/+32
2021-07-20	net/sched: act_skbmod: Skip non-Ethernet packets	Peilin Ye	1	-4/+8
2021-07-16	netdevsim: Add multi-queue support	Peilin Ye	3	-9/+15
2021-07-15	Merge branch 'sockmap: add sockmap support for unix datagram socket'	Alexei Starovoitov	11	-128/+564
2021-07-15	selftests/bpf: Add test cases for redirection between udp and unix	Cong Wang	1	-0/+170
2021-07-15	selftests/bpf: Add a test case for unix sockmap	Cong Wang	1	-0/+97
2021-07-15	selftests/bpf: Factor out add_to_sockmap()	Cong Wang	1	-38/+21
2021-07-15	selftests/bpf: Factor out udp_socketpair()	Cong Wang	1	-38/+40
2021-07-15	af_unix: Implement unix_dgram_bpf_recvmsg()	Cong Wang	3	-3/+93
2021-07-15	af_unix: Implement ->psock_update_sk_prot()	Cong Wang	6	-1/+65
2021-07-15	af_unix: Add a dummy ->close() for sockmap	Cong Wang	1	-0/+9
2021-07-15	af_unix: Set TCP_ESTABLISHED for datagram sockets too	Cong Wang	1	-6/+8
2021-07-15	af_unix: Implement ->read_sock() for sockmap	Cong Wang	1	-0/+37
2021-07-15	sock_map: Lift socket state restriction for datagram sockets	Cong Wang	3	-24/+8
2021-07-15	sock_map: Relax config dependency to CONFIG_NET	Cong Wang	3	-21/+21
2021-07-15	bpf, sockmap, udp: sk_prot needs inuse_idx set for proc stats	Jakub Sitnicki	1	-1/+1
2021-07-15	bpf, sockmap, tcp: sk_prot needs inuse_idx set for proc stats	John Fastabend	1	-1/+1
2021-07-15	bpf, sockmap: Fix potential memory leak on unlikely error case	John Fastabend	1	-5/+11
2021-07-15	net_sched: introduce tracepoint trace_qdisc_enqueue()	Qitao Xu	2	-4/+42
2021-07-15	net_sched: use %px to print skb address in trace_qdisc_dequeue()	Qitao Xu	1	-1/+1
2021-07-15	net: use %px to print skb address in trace_netif_receive_skb	Qitao Xu	1	-1/+1
2021-06-28	net: sched: fix warning in tcindex_alloc_perfect_hash	Pavel Skripkin	1	-1/+1
2021-06-23	Merge git://git.kernel.org/pub/scm/linux/kernel/git/bpf/bpf	David S. Miller	13	-64/+137
2021-06-21	skmsg: Increase sk->sk_drops when dropping packets	Cong Wang	1	-8/+14
2021-06-21	skmsg: Pass source psock to sk_psock_skb_redirect()	Cong Wang	1	-5/+6
2021-06-21	skmsg: Teach sk_psock_verdict_apply() to return errors	Cong Wang	1	-9/+14
2021-06-21	skmsg: Fix a memory leak in sk_psock_verdict_apply()	Cong Wang	1	-0/+5
2021-06-21	skmsg: Clear skb redirect pointer before dropping it	Cong Wang	1	-0/+2
2021-06-21	udp: Fix a memory leak in udp_read_sock()	Cong Wang	1	-0/+2
2021-06-21	selftests/bpf: Retry for EAGAIN in udp_redir_to_connected()	Cong Wang	1	-1/+6
2021-06-21	skmsg: Improve udp_bpf_recvmsg() accuracy	Cong Wang	4	-31/+65
2021-06-12	net: make get_net_ns return error if NET_NS is disabled	Changbin Du	4	-15/+19
2021-05-25	net: zero-initialize tc skb extension on allocation	Vlad Buslov	4	-3/+14
2021-05-23	MAINTAINERS: Add entries for CBS, ETF and taprio qdiscs	Vinicius Costa Gomes	1	-0/+8
2021-05-18	skmsg: Remove unused parameters of sk_msg_wait_data()	Cong Wang	4	-17/+6
2021-05-10	rtnetlink: avoid RCU read lock when holding RTNL	Cong Wang	2	-21/+9
2021-05-05	smc: disallow TCP_ULP in smc_setsockopt()	Cong Wang	1	-1/+3
2021-05-05	net: fix nla_strcmp to handle more then one trailing null character	Maciej Żenczykowski	1	-1/+1
2021-04-29	net/sched: sch_frag: fix stack OOB read while fragmenting IPv4 packets	Davide Caratti	1	-4/+4
2021-04-27	net: fix a concurrency bug in l2tp_tunnel_register()	Gong, Sishuai	1	-5/+5
2021-04-19	net: sched: tapr: prevent cycle_time == 0 in parse_taprio_schedule	Du Cheng	1	-0/+6
2021-04-16	netlink: don't call ->netlink_bind with table lock held	Florian Westphal	1	-2/+2
2021-04-12	sock_map: Fix a potential use-after-free in sock_map_close()	Cong Wang	1	-1/+2
2021-04-12	skmsg: Pass psock pointer to ->psock_update_sk_prot()	Cong Wang	7	-10/+12
2021-04-08	Merge branch 'net-sched-action-tests'	David S. Miller	1	-0/+59
2021-04-08	net: sched: fix err handler in tcf_action_init()	Vlad Buslov	3	-19/+19
2021-04-07	bpf, sockmap: Fix sk->prot unhash op reset	John Fastabend	1	-1/+6
2021-04-06	bpf, udp: Remove some pointless comments	Cong Wang	1	-2/+0
2021-04-01	Merge branch 'sockmap: introduce BPF_SK_SKB_VERDICT and support UDP'	Alexei Starovoitov	26	-237/+677
2021-04-01	selftests/bpf: Add a test case for loading BPF_SK_SKB_VERDICT	Cong Wang	2	-0/+58
2021-04-01	selftests/bpf: Add a test case for udp sockmap	Cong Wang	2	-0/+158
2021-04-01	sock_map: Update sock type checks for UDP	Cong Wang	1	-1/+4
2021-04-01	udp: Implement udp_bpf_recvmsg() for sockmap	Cong Wang	1	-1/+63
2021-04-01	skmsg: Extract __tcp_bpf_recvmsg() and tcp_bpf_wait_data()	Cong Wang	5	-102/+106
2021-04-01	udp: Implement ->read_sock() for sockmap	Cong Wang	4	-0/+33
2021-04-01	sock: Introduce sk->sk_prot->psock_update_sk_prot()	Cong Wang	12	-45/+58
2021-04-01	sock_map: Introduce BPF_SK_SKB_VERDICT	Cong Wang	8	-1/+38
2021-04-01	sock_map: Kill sock_map_link_no_progs()	Cong Wang	1	-40/+15
2021-04-01	sock_map: Simplify sock_map_link() a bit	Cong Wang	1	-4/+5
2021-04-01	skmsg: Use GFP_KERNEL in sk_psock_create_ingress_msg()	Cong Wang	1	-1/+1
2021-04-01	skmsg: Use rcu work for destroying psock	Cong Wang	2	-16/+6
2021-04-01	skmsg: Avoid lock_sock() in sk_psock_backlog()	Cong Wang	3	-16/+37
2021-04-01	net: Introduce skb_send_sock() for sock_map	Cong Wang	2	-7/+49
2021-04-01	skmsg: Introduce a spinlock to protect ingress_msg	Cong Wang	3	-12/+55
2021-04-01	skmsg: Lock ingress_skb when purging	Cong Wang	1	-1/+1
2021-03-29	bpf: tcp: Limit calling some tcp cc functions to CONFIG_DYNAMIC_FTRACE	Martin KaFai Lau	1	-0/+2
2021-03-26	Merge branch 'mld-sleepable'	David S. Miller	9	-471/+678
2021-03-26	mld: add mc_lock for protecting per-interface mld data	Taehee Yoo	2	-116/+194
2021-03-26	mld: add new workqueues for process mld events	Taehee Yoo	4	-86/+210
2021-03-26	mld: convert ifmcaddr6 to RCU	Taehee Yoo	7	-188/+140
2021-03-26	mld: convert ip6_sf_list to RCU	Taehee Yoo	2	-77/+130
2021-03-26	mld: convert ipv6_mc_socklist->sflist to RCU	Taehee Yoo	2	-32/+24
2021-03-26	mld: get rid of inet6_dev->mc_lock	Taehee Yoo	2	-10/+0
2021-03-26	mld: convert from timer to delayed work	Taehee Yoo	2	-65/+83
2021-03-11	Merge branch 'htb-fixes'	David S. Miller	1	-6/+13
2021-03-09	Merge git://git.kernel.org/pub/scm/linux/kernel/git/netdev/net	Linus Torvalds	159	-786/+1456
2021-03-04	skmsg: Add function doc for skb->_sk_redir	Cong Wang	1	-0/+1
2021-02-26	Merge branch 'sock_map: clean up and refactor code for BPF_SK_SKB_VERDICT'	Alexei Starovoitov	16	-259/+274
2021-02-26	skmsg: Remove unused sk_psock_stop() declaration	Cong Wang	1	-1/+0
2021-02-26	skmsg: Get rid of sk_psock_bpf_run()	Cong Wang	1	-10/+4
2021-02-26	skmsg: Make __sk_psock_purge_ingress_msg() static	Cong Wang	2	-3/+1
2021-02-26	sock_map: Make sock_map_prog_update() static	Cong Wang	2	-11/+5
2021-02-26	sock_map: Rename skb_parser and skb_verdict	Cong Wang	5	-47/+47
2021-02-26	skmsg: Move sk_redir from TCP_SKB_CB to skb	Cong Wang	5	-37/+62
2021-02-26	bpf: Compute data_end dynamically with JIT code	Cong Wang	3	-27/+28
2021-02-26	skmsg: Get rid of struct sk_psock_parser	Cong Wang	3	-53/+27
2021-02-26	bpf: Clean up sockmap related Kconfigs	Cong Wang	12	-103/+133
2021-02-22	bpf: Clear percpu pointers in bpf_prog_clone_free()	Cong Wang	1	-0/+2
2021-02-16	net: sched: fix police ext initialization	Vlad Buslov	3	-1/+3
2021-02-11	net: fix dev_ifsioc_locked() race condition	Cong Wang	6	-21/+58
2021-02-01	net: sched: replaced invalid qdisc tree flush helper in qdisc_replace	Alexander Ovechkin	1	-1/+1
2021-01-30	neighbour: Prevent a dead entry from updating gc_list	Chinmay Agarwal	1	-3/+4
2021-01-28	skmsg: Make sk_psock_destroy() static	Cong Wang	2	-3/+1
2021-01-26	team: protect features update by RCU to avoid deadlock	Ivan Vecera	1	-3/+3
2021-01-21	Merge branch 'master' of git://git.kernel.org/pub/scm/linux/kernel/git/klasse...	Jakub Kicinski	4	-15/+68
2021-01-18	net_sched: fix RTNL deadlock again caused by request_module()	Cong Wang	3	-41/+79
2021-01-15	net_sched: reject silly cell_log in qdisc_get_rtab()	Eric Dumazet	1	-1/+2

[next]

index : kernel/git/netdev/net-next.git

Netdev Group's -next networking tree

main switch

Netdev Group

about summary refs log tree commit diff stats

log msg Cong Wang search

Age	Commit message (Expand)	Author	Files	Lines
2021-01-15	cls_flower: call nla_ok() before nla_next()	Cong Wang	1	-9/+13
2021-01-13	can: isotp: isotp_getname(): fix kernel information leak	Oliver Hartkopp	1	-0/+1
2021-01-11	net: bareudp: simplify error paths calling dellink	Jakub Kicinski	1	-6/+2
2021-01-04	af_key: relax availability checks for skb size calculation	Cong Wang	1	-3/+3
2020-12-28	erspan: fix version 1 check in gre_parse_header()	Cong Wang	1	-1/+1
2020-12-28	net: sched: prevent invalid Scell_log shift count	Randy Dunlap	5	-5/+7
2020-12-10	Merge git://git.kernel.org/pub/scm/linux/kernel/git/bpf/bpf	David S. Miller	21	-88/+163
2020-12-07	lwt_bpf: Replace preempt_disable() with migrate_disable()	Cong Wang	1	-3/+3
2020-12-07	lwt: Disable BH too in run_lwt_bpf()	Dongdong Wang	1	-4/+4
2020-12-04	net/sched: fq_pie: initialize timer earlier in fq_pie_init()	Davide Caratti	1	-1/+1
2020-12-01	iommu: avoid taking iova_rbtree_lock twice	Cong Wang	1	-2/+6
2020-11-27	net/sched: sch_frag: add generic packet fragment support.	wenxu	7	-1/+179
2020-11-05	net: sched: implement action-specific terse dump	Vlad Buslov	2	-32/+41
2020-10-29	tipc: fix memory leak caused by tipc_buf_append()	Tung Nguyen	1	-3/+2
2020-10-20	net/sched: act_tunnel_key: fix OOB write in case of IPv6 ERSPAN tunnels	Davide Caratti	1	-1/+1
2020-10-20	net/sched: act_gate: Unlock ->tcfa_lock in tc_setup_flow_action()	Guillaume Nault	1	-1/+1
2020-10-15	net: sched: Fix suspicious RCU usage while accessing tcf_tunnel_info	Leon Romanovsky	1	-1/+4
2020-10-13	ip_gre: set dev->hard_header_len and dev->needed_headroom properly	Cong Wang	1	-4/+11
2020-10-10	Merge tag 'linux-can-fixes-for-5.9-20201008' of git://git.kernel.org/pub/scm/...	Jakub Kicinski	2	-2/+2
2020-10-09	tipc: fix the skb_unshare() in tipc_buf_append()	Cong Wang	1	-1/+2
2020-10-09	net/sched: get rid of qdisc->padded	Eric Dumazet	3	-23/+10
2020-10-08	can: j1935: j1939_tp_tx_dat_new(): fix missing initialization of skbcnt	Cong Wang	1	-0/+1
2020-10-04	net_sched: check error pointer in tcf_dump_walker()	Cong Wang	1	-0/+2
2020-10-02	scsi: libiscsi: use sendpage_ok() in iscsi_tcp_segment_map()	Coly Li	1	-1/+1
2020-10-02	net: add WARN_ONCE in kernel_sendpage() for improper zero-copy send	Coly Li	1	-2/+4
2020-09-28	net_sched: remove a redundant goto chain check	Cong Wang	1	-14/+0
2020-09-24	Merge branch 'net_sched-fix-a-UAF-in-tcf_action_init'	David S. Miller	22	-66/+35
2020-09-24	net_sched: commit action insertions together	Cong Wang	1	-9/+23
2020-09-24	net_sched: defer tcf_idr_insert() in tcf_action_init_1()	Cong Wang	22	-66/+21
2020-09-14	net/packet: Fix a comment about hard_header_len and headroom allocation	Xie He	1	-6/+9
2020-09-08	net: dsa: link interfaces with the DSA master to get rid of lockdep warnings	Vladimir Oltean	1	-2/+16
2020-09-04	act_ife: load meta modules before tcf_idr_check_alloc()	Cong Wang	1	-10/+34
2020-09-01	hostap: Remove unused variable 'fc'	Lee Jones	1	-2/+1
2020-08-28	net_sched: fix error path in red_init()	Cong Wang	1	-16/+4
2020-08-17	Merge git://git.kernel.org/pub/scm/linux/kernel/git/netdev/net	Linus Torvalds	76	-349/+768
2020-08-15	tipc: fix uninit skb->data in tipc_nl_compat_dumpit()	Cong Wang	1	-1/+11
2020-08-16	bonding: fix a potential double-unregister	Cong Wang	1	-1/+2
2020-08-13	net: openvswitch: introduce common code for flushing flows	Tonghao Zhang	3	-21/+27
2020-08-03	net/sched: act_ct: fix miss set mru for ovs after defrag in act_ct	wenxu	5	-3/+11
2020-08-01	Merge git://git.kernel.org/pub/scm/linux/kernel/git/netdev/net	Linus Torvalds	81	-451/+782
2020-07-30	ipv6: fix memory leaks on IPV6_ADDRFORM path	Cong Wang	3	-5/+14
2020-07-28	net_sched: initialize timer earlier in red_init()	Cong Wang	1	-4/+4
2020-07-25	Merge git://git.kernel.org/pub/scm/linux/kernel/git/netdev/net into master	Linus Torvalds	127	-676/+1022
2020-07-24	qrtr: orphan socket in qrtr_release()	Cong Wang	1	-0/+1
2020-07-23	geneve: fix an uninitialized value in geneve_changelink()	Cong Wang	1	-1/+1
2020-07-23	bonding: check return value of register_netdevice() in bond_newlink()	Cong Wang	1	-2/+1
2020-07-13	net: sched: kerneldoc fixes	Andrew Lunn	2	-2/+2
2020-07-10	Merge git://git.kernel.org/pub/scm/linux/kernel/git/netdev/net	Linus Torvalds	197	-922/+1634
2020-07-09	net_sched: fix a memory leak in atm_tc_init()	Cong Wang	1	-4/+4
2020-07-07	cgroup: fix cgroup_sk_alloc() for sk_clone_lock()	Cong Wang	4	-16/+27
2020-07-01	genetlink: remove genl_bind	Sean Tranchetti	2	-57/+0
2020-06-29	genetlink: get rid of family->attrbuf	Cong Wang	2	-37/+13
2020-06-28	net: explain the lockdep annotations for dev_uc_unsync()	Cong Wang	1	-0/+10
2020-06-28	net: get rid of lockdep_set_class_and_subclass()	Cong Wang	3	-11/+8
2020-06-18	net/sched: act_gate: fix configuration of the periodic timer	Davide Caratti	1	-35/+55
2020-06-13	Merge git://git.kernel.org/pub/scm/linux/kernel/git/netdev/net	Linus Torvalds	111	-647/+1344
2020-06-12	genetlink: clean up family attributes allocations	Cong Wang	1	-16/+12
2020-06-09	net: change addr_list_lock back to static key	Cong Wang	17	-36/+76
2020-06-04	genetlink: fix memory leaks in genl_family_rcv_msg_dumpit()	Cong Wang	1	-36/+58
2020-06-03	Merge git://git.kernel.org/pub/scm/linux/kernel/git/netdev/net-next	Linus Torvalds	2097	-44648/+150201
2020-06-02	mm/migrate.c: call detach_page_private to cleanup code	Guoqing Jiang	1	-6/+1
2020-06-02	iomap: convert from readpages to readahead	Matthew Wilcox (Oracle)	5	-74/+41
2020-06-02	fuse: convert from readpages to readahead	Matthew Wilcox (Oracle)	1	-72/+28
2020-06-02	f2fs: pass the inode to f2fs_mpage_readpages	Matthew Wilcox (Oracle)	1	-4/+3
2020-06-02	f2fs: convert from readpages to readahead	Matthew Wilcox (Oracle)	2	-31/+22
2020-06-02	ext4: pass the inode to ext4_mpage_readpages	Matthew Wilcox (Oracle)	3	-5/+4
2020-06-02	ext4: convert from readpages to readahead	Matthew Wilcox (Oracle)	3	-26/+18
2020-06-02	erofs: convert compressed files from readpages to readahead	Matthew Wilcox (Oracle)	1	-20/+9
2020-06-02	erofs: convert uncompressed files from readpages to readahead	Matthew Wilcox (Oracle)	3	-29/+18
2020-06-02	btrfs: convert from readpages to readahead	Matthew Wilcox (Oracle)	3	-42/+20
2020-06-02	fs: convert mpage_readpages to mpage_readahead	Matthew Wilcox (Oracle)	18	-126/+73
2020-06-02	mm: use memalloc_nofs_save in readahead path	Matthew Wilcox (Oracle)	1	-0/+14
2020-06-02	mm: document why we don't set PageReadahead	Matthew Wilcox (Oracle)	1	-3/+6
2020-06-02	mm: add page_cache_readahead_unbounded	Matthew Wilcox (Oracle)	6	-91/+55
2020-06-02	mm: move end_index check out of readahead loop	Matthew Wilcox (Oracle)	1	-6/+8
2020-06-02	mm: add readahead address space operation	Matthew Wilcox (Oracle)	4	-3/+32
2020-06-02	mm: put readahead pages in cache earlier	Matthew Wilcox (Oracle)	1	-18/+28
2020-06-02	mm: remove 'page_offset' from readahead loop	Matthew Wilcox (Oracle)	1	-5/+3
2020-06-02	mm: rename readahead loop variable to 'i'	Matthew Wilcox (Oracle)	1	-4/+4
2020-06-02	mm: rename various 'offset' parameters to 'index'	Matthew Wilcox (Oracle)	1	-44/+42
2020-06-02	mm: use readahead_control to pass arguments	Matthew Wilcox (Oracle)	1	-14/+19
2020-06-02	mm: add new readahead_control API	Matthew Wilcox (Oracle)	1	-0/+140
2020-06-02	mm: move readahead nr_pages check into read_pages	Matthew Wilcox (Oracle)	1	-5/+7
2020-06-02	mm: ignore return value of ->readpages	Matthew Wilcox (Oracle)	1	-6/+2
2020-06-02	mm: return void from various readahead functions	Matthew Wilcox (Oracle)	3	-28/+19
2020-06-02	mm: move readahead prototypes from mm.h	Matthew Wilcox (Oracle)	5	-19/+13
2020-06-01	flow_dissector: work around stack frame size warning	Arnd Bergmann	1	-9/+8
2020-05-30	devinet: fix memleak in inetdev_init()	Yang Yingliang	1	-0/+1
2020-05-28	perf metricgroup: Remove unnecessary ',' from events	Ian Rogers	1	-2/+7
2020-05-28	perf metricgroup: Add options to not group or merge	Ian Rogers	5	-19/+87
2020-05-28	perf metricgroup: Remove duped metric group events	Ian Rogers	1	-29/+62
2020-05-28	perf metricgroup: Order event groups by size	Ian Rogers	1	-1/+15
2020-05-28	perf metricgroup: Delay events string creation	Ian Rogers	1	-12/+21
2020-05-28	perf metricgroup: Use early return in add_metric	Ian Rogers	1	-7/+15
2020-05-28	perf metricgroup: Always place duration_time last	Ian Rogers	1	-9/+9
2020-05-28	perf metricgroup: Free metric_events on error	Ian Rogers	1	-0/+3
2020-05-28	perf metricgroup: Make 'evlist_used' variable a bitmap instead of array of bools	Ian Rogers	1	-10/+8
2020-05-28	perf expr: Migrate expr ids table to a hashmap	Ian Rogers	7	-190/+200
2020-05-28	perf tools: Grab a copy of libbpf's hashmap	Ian Rogers	4	-0/+422
2020-05-27	Merge branch 'net_sched-reduce-the-number-of-qdisc-resets'	David S. Miller	3	-43/+110
2020-05-27	net_sched: get rid of unnecessary dev_qdisc_reset()	Cong Wang	1	-16/+0
2020-05-27	net_sched: avoid resetting active qdisc for multiple times	Cong Wang	1	-14/+23
2020-05-27	net_sched: add a tracepoint for qdisc creation	Cong Wang	3	-1/+29
2020-05-27	net_sched: add tracepoints for qdisc_reset() and qdisc_destroy()	Cong Wang	2	-0/+56
2020-05-27	net_sched: use qdisc_reset() in qdisc_destroy()	Cong Wang	1	-13/+3
2020-05-23	net: psample: fix build error when CONFIG_INET is not enabled	Randy Dunlap	1	-0/+8
2020-05-07	net: fix a potential recursive NETDEV_FEAT_CHANGE	Cong Wang	1	-1/+3
2020-05-06	Merge git://git.kernel.org/pub/scm/linux/kernel/git/netdev/net	Linus Torvalds	107	-462/+801
2020-05-05	sch_choke: Remove classid from choke_skb_cb.	David S. Miller	1	-1/+0
2020-05-04	net: partially revert dynamic lockdep key changes	Cong Wang	22	-33/+294
2020-05-04	Merge branch 'net-reduce-dynamic-lockdep-keys'	David S. Miller	23	-37/+294
2020-05-04	bonding: remove useless stats_lock_key	Cong Wang	2	-4/+0
2020-05-04	atm: fix a memory leak of vcc->user_back	Cong Wang	1	-0/+6
2020-05-04	atm: fix a UAF in lec_arp_clear_vccs()	Cong Wang	1	-11/+11
2020-05-04	net_sched: fix tcm_parent in tc filter dump	Cong Wang	2	-4/+5
2020-05-01	Merge branch 'Introduce-a-flow-gate-control-action-and-apply-IEEE'	David S. Miller	13	-1/+2268
2020-04-27	sch_choke: avoid potential panic in choke_reset()	Eric Dumazet	1	-1/+2
2020-04-07	Merge git://git.kernel.org/pub/scm/linux/kernel/git/netdev/net	Linus Torvalds	43	-163/+361
2020-04-03	net_sched: fix a missing refcnt in tcindex_init()	Cong Wang	1	-0/+1
2020-04-01	net_sched: add a temporary refcnt for struct tcindex_data	Cong Wang	1	-6/+38
2020-03-25	Merge git://git.kernel.org/pub/scm/linux/kernel/git/netdev/net	Linus Torvalds	159	-945/+2140
2020-03-16	net_sched: cls_route: remove the right filter from hashtable	Cong Wang	1	-2/+2
2020-03-14	net_sched: keep alloc_hash updated after hash allocation	Cong Wang	1	-0/+1
2020-03-14	net_sched: hold rtnl lock in tcindex_partial_destroy_work()	Cong Wang	1	-0/+2
2020-03-12	Revert "net: sched: make newly activated qdiscs visible"	Julian Wiedmann	3	-28/+0
2020-03-11	net: sched: make newly activated qdiscs visible	Julian Wiedmann	3	-0/+28
2020-02-27	dma-buf: free dmabuf->name in dma_buf_release()	Cong Wang	1	-0/+1
2020-02-26	Merge git://git.kernel.org/pub/scm/linux/kernel/git/pablo/nf	David S. Miller	6	-228/+529
2020-02-26	netfilter: xt_hashlimit: unregister proc file before releasing mutex	Cong Wang	1	-10/+6
2020-02-21	Merge git://git.kernel.org/pub/scm/linux/kernel/git/netdev/net	Linus Torvalds	104	-506/+1132
2020-02-18	Merge git://git.kernel.org/pub/scm/linux/kernel/git/pablo/nf	David S. Miller	7	-45/+220
2020-02-17	net: sched: correct flower port blocking	Jason Baron	2	-0/+10
2020-02-10	perf symbols: Convert symbol__is_idle() to use strlist	Kim Phillips	1	-5/+9
2020-02-10	perf symbols: Update the list of kernel idle symbols	Kim Phillips	1	-0/+3
2020-02-10	perf stat: Don't report a null stalled cycles per insn metric	Kim Phillips	1	-6/+0
2020-02-07	netfilter: xt_hashlimit: limit the max size of hashtable	Cong Wang	1	-0/+10
2020-02-07	netfilter: xt_hashlimit: reduce hashlimit_mutex scope for htable_put()	Cong Wang	1	-6/+6
2020-02-06	net: sched: prevent a use after free	Dan Carpenter	1	-1/+1
2020-02-05	net_sched: fix a resource leak in tcindex_set_parms()	Cong Wang	1	-2/+1
2020-02-04	Merge git://git.kernel.org/pub/scm/linux/kernel/git/netdev/net	Linus Torvalds	47	-246/+569
2020-02-04	net_sched: fix an OOB access in cls_tcindex	Cong Wang	1	-20/+20
2020-02-01	cls_rsvp: fix rsvp_policy	Eric Dumazet	1	-4/+2
2020-01-27	net_sched: ematch: reject invalid TCF_EM_SIMPLE	Eric Dumazet	1	-0/+3
2020-01-27	net_sched: walk through all child classes in tc_bind_tclass()	Cong Wang	1	-11/+30
2020-01-27	net_sched: fix ops->bind_class() implementations	Cong Wang	12	-44/+97
2020-01-23	net_sched: fix datalen for ematch	Cong Wang	1	-1/+1
2020-01-22	net_sched: use validated TCA_KIND attribute in tc_new_tfilter()	Eric Dumazet	1	-3/+2
2020-01-19	Merge git://git.kernel.org/pub/scm/linux/kernel/git/netdev/net	Linus Torvalds	92	-377/+773
2020-01-19	net: sched: act_ctinfo: fix memory leak	Eric Dumazet	1	-0/+11
2020-01-17	net: avoid updating qdisc_xmit_lock_key in netdev_update_lockdep_key()	Cong Wang	1	-12/+0
2020-01-17	net/sched: act_ife: initalize ife->metalist earlier	Eric Dumazet	1	-4/+3
2020-01-17	Merge git://git.kernel.org/pub/scm/linux/kernel/git/pablo/nf	David S. Miller	5	-24/+54
2020-01-13	netfilter: fix a use-after-free in mtype_destroy()	Cong Wang	1	-1/+1
2020-01-02	sch_cake: avoid possible divide by zero in cake_enqueue()	Wen Yang	1	-1/+1
2019-12-30	netfilter: arp_tables: init netns pointer in xt_tgchk_param struct	Florian Westphal	1	-11/+16
2019-12-10	tracing: remove set but not used variable 'buffer'	YueHaibing	1	-2/+0
2019-12-07	net_sched: validate TCA_KIND attribute in tc_chain_tmplt_add()	Eric Dumazet	1	-1/+7
2019-12-07	gre: refetch erspan header from skb->data after pskb_may_pull()	Cong Wang	1	-1/+1
2019-12-05	hsr: fix a NULL pointer dereference in hsr_dev_xmit()	Taehee Yoo	1	-2/+7
2019-12-02	tracing: Introduce trace event injection	Cong Wang	5	-0/+348
2019-11-30	net: sched: fix `tc -s class show` no bstats on class with nolock subqueues	Dust Li	4	-5/+6
2019-10-30	net/sched: Replace rcu_swap_protected() with rcu_replace_pointer()	Paul E. McKenney	12	-20/+21
2019-10-17	net: sched: Avoid using yield() in a busy waiting loop	Marc Kleine-Budde	1	-2/+7
2019-10-08	net_sched: fix backward compatibility for TCA_ACT_KIND	Cong Wang	1	-4/+5
2019-10-08	net_sched: fix backward compatibility for TCA_KIND	Cong Wang	2	-5/+34
2019-09-27	Merge branch 'qdisc-destroy'	David S. Miller	3	-11/+23
2019-09-27	net: sched: sch_sfb: don't call qdisc_put() while holding tree lock	Vlad Buslov	1	-3/+4
2019-09-26	net/sched: cbs: Fix not adding cbs instance to list	Vinicius Costa Gomes	1	-17/+13
2019-09-21	net_sched: add policy validation for action attributes	Cong Wang	1	-16/+18
2019-09-21	net_sched: add max len check for TCA_KIND	Cong Wang	1	-1/+2
2019-09-15	net_sched: let qdisc_put() accept NULL pointer	Cong Wang	1	-0/+3
2019-09-10	sch_hhf: ensure quantum and hhf_non_hh_weight are non-zero	Cong Wang	1	-1/+1
2019-09-10	net_sched: check cops->tcf_block in tc_bind_tclass()	Cong Wang	1	-0/+2
2019-09-06	net_sched: act_police: add 2 new attributes to support police 64bit rate and ...	David Dai	2	-4/+25
2019-08-27	net_sched: fix a NULL pointer deref in ipt action	Cong Wang	20	-24/+27
2019-08-08	sched/fair: Fix low cpu usage with high throttling by removing expiration of ...	Dave Chiluk	3	-63/+67
2019-08-05	net_sched: use temporary variable for actions indexes	Dmytro Linkin	18	-75/+100
2019-07-25	ife: error out when nla attributes are empty	Cong Wang	1	-0/+5
2019-07-24	netrom: hold sock when setting skb->destructor	Cong Wang	1	-0/+1
2019-07-23	Merge tag 'perf-urgent-for-mingo-5.3-20190723' of git://git.kernel.org/pub/sc...	Ingo Molnar	12	-19/+54
2019-07-23	perf stat: Always separate stalled cycles per insn	Cong Wang	1	-1/+2
2019-07-19	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	102	-400/+965
2019-07-17	Merge branch 'ipv4-relax-source-validation-check-for-loopback-packets'	David S. Miller	2	-1/+39
2019-07-17	selftests: add a test case for rp_filter	Cong Wang	1	-1/+34
2019-07-17	fib: relax source validation check for loopback packets	Cong Wang	1	-0/+5
2019-07-17	net_sched: unset TCQ_F_CAN_BYPASS when adding filters	Cong Wang	3	-4/+1
2019-07-16	tracing: Make trace_get_fields() global	Cong Wang	2	-6/+8
2019-07-16	tracing: Let filter_assign_type() detect FILTER_PTR_STRING	Cong Wang	1	-0/+3
2019-07-16	tracing: Pass type into tracing_generic_entry_update()	Cong Wang	3	-6/+6
2019-07-11	hsr: switch ->dellink() to ->ndo_uninit()	Cong Wang	3	-18/+8
2019-07-08	Merge branch 'Add-MPLS-actions-to-TC'	David S. Miller	11	-73/+1754
2019-07-08	net: sched: add mpls manipulation actions to TC	John Hurley	8	-1/+514
2019-07-08	net: core: add MPLS update core helper and use in OvS	John Hurley	3	-10/+37
2019-07-08	net: core: move pop MPLS functionality from OvS to core helper	John Hurley	3	-35/+45
2019-07-08	net: core: move push MPLS functionality from OvS to core helper	John Hurley	3	-27/+69
2019-07-08	skbuff: increase verbosity when dumping skb data	Willem de Bruijn	3	-12/+104
2019-07-05	Merge branch 'hsr-bug-fixes'	David S. Miller	6	-15/+37
2019-07-05	hsr: fix a NULL pointer deref in hsr_dev_xmit()	Cong Wang	1	-3/+7
2019-07-05	hsr: implement dellink to clean up resources	Cong Wang	5	-10/+25
2019-07-05	hsr: fix a memory leak in hsr_del_port()	Cong Wang	2	-2/+5

[prev] [next]

index : kernel/git/netdev/net-next.git

Netdev Group's -next networking tree

main switch

Netdev Group

about summary refs log tree commit diff stats

log msg Cong Wang search

Age	Commit message (Expand)	Author	Files	Lines
2019-07-05	Merge branch 'master' of git://git.kernel.org/pub/scm/linux/kernel/git/klasse...	David S. Miller	6	-81/+86
2019-07-03	bonding: validate ip header before check IPPROTO_IGMP	Cong Wang	1	-14/+23
2019-07-02	xfrm: remove a duplicated assignment	Cong Wang	1	-3/+0
2019-07-01	Merge branch 'idr-fix-overflow-cases-on-32-bit-CPU'	David S. Miller	5	-29/+57
2019-07-01	selftests: add a test case for cls_lower handle overflow	Davide Caratti	1	-0/+19
2019-07-01	idr: introduce idr_for_each_entry_continue_ul()	Cong Wang	3	-24/+27
2019-07-01	idr: fix overflow case for idr_for_each_entry_ul()	Cong Wang	2	-5/+11
2019-07-01	netrom: fix a memory leak in nr_rx_frame()	Cong Wang	1	-1/+2
2019-06-18	net: netem: fix use after free and double free with packet corruption	Jakub Kicinski	1	-8/+7
2019-06-18	net: netem: fix backlog accounting for corrupted GSO frames	Jakub Kicinski	1	-5/+8
2019-06-07	RAS/CEC: Convert the timer callback to a workqueue	Cong Wang	1	-24/+22
2019-06-07	RAS/CEC: Fix binary search function	Borislav Petkov	1	-14/+20
2019-05-29	net: sched: Introduce act_ctinfo action	Kevin 'ldir' Darbyshire-Bryant	7	-0/+478
2019-05-09	tuntap: synchronize through tfiles array instead of tun->numqueues	Jason Wang	1	-1/+6
2019-05-07	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net-next	Linus Torvalds	1636	-26976/+126809
2019-05-07	taprio: add null check on sched_nest to avoid potential null pointer dereference	Colin Ian King	1	-0/+2
2019-05-05	sch_htb: redefine htb qdisc overlimits	Cong Wang	1	-3/+6
2019-05-05	cls_cgroup: avoid panic when receiving a packet before filter set	Matteo Croce	1	-0/+2
2019-05-04	cls_matchall: avoid panic when receiving a packet before filter set	Matteo Croce	1	-0/+3
2019-05-04	net: add a generic tracepoint for TX queue timeout	Cong Wang	2	-0/+25
2019-04-30	Merge branch 'master' of git://git.kernel.org/pub/scm/linux/kernel/git/klasse...	David S. Miller	13	-49/+89
2019-04-22	net: tc_act: drop include of module.h from tc_ife.h	Paul Gortmaker	1	-1/+2
2019-04-10	Merge branch 'net-sched-taprio-fix-picos_per_byte-miscalculation'	David S. Miller	2	-30/+165
2019-04-09	net/mlx5e: Rx, Check ip headers sanity	Saeed Mahameed	1	-1/+8
2019-03-26	xfrm: clean up xfrm protocol checks	Cong Wang	5	-16/+23
2019-03-23	net: sched: fix cleanup NULL pointer exception in act_mirr	John Hurley	1	-2/+5
2019-03-21	Merge branch 'net-sched-validate-the-control-action-with-all-the-other-parame...	David S. Miller	36	-121/+749
2019-03-21	net/sched: let actions use RCU to access 'goto_chain'	Davide Caratti	4	-10/+13
2019-03-13	net_sched: return correct value for "notify" functions	Zhike Wang	2	-13/+34
2019-03-05	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net-next	Linus Torvalds	2150	-57877/+112531
2019-03-02	net: sched: put back q.qlen into a single location	Eric Dumazet	3	-27/+19
2019-02-26	net: sched: fix typo in walker_check_empty()	Vlad Buslov	1	-2/+2
2019-02-25	net: socket: set sock->sk to NULL after calling proto_ops::release()	Eric Biggers	1	-0/+1
2019-02-25	net: sched: don't release block->lock when dumping chains	Vlad Buslov	1	-9/+7
2019-02-25	net: sched: set dedicated tcf_walker flag when tp is empty	Vlad Buslov	2	-4/+10
2019-02-24	Merge branch 'Add-tests-for-unlocked-flower-classifier-implementation'	David S. Miller	4	-4/+297
2019-02-22	net_sched: initialize net pointer inside tcf_exts_init()	Cong Wang	13	-29/+30
2019-02-21	Merge branch 'master' of git://git.kernel.org/pub/scm/linux/kernel/git/klasse...	David S. Miller	9	-47/+53
2019-02-20	net_sched: fix a memory leak in cls_tcindex	Cong Wang	1	-13/+24
2019-02-20	net_sched: fix a race condition in tcindex_destroy()	Cong Wang	1	-7/+11
2019-02-15	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	67	-220/+429
2019-02-12	team: avoid complex list operations in team_nl_cmd_options_set()	Cong Wang	1	-22/+5
2019-02-12	Merge branch 'net_sched-some-fixes-for-cls_tcindex'	David S. Miller	1	-32/+48
2019-02-12	net_sched: fix two more memory leaks in cls_tcindex	Cong Wang	1	-9/+7
2019-02-12	net_sched: fix a memory leak in cls_tcindex	Cong Wang	1	-16/+30
2019-02-12	net_sched: fix a race condition in tcindex_destroy()	Cong Wang	1	-7/+11
2019-02-11	mlx5: use RCU lock in mlx5_eq_cq_get()	Cong Wang	1	-6/+6
2019-02-05	xfrm: destroy xfrm_state synchronously on net exit path	Cong Wang	5	-17/+31
2019-01-31	rds: fix refcount bug in rds_sock_addref	Eric Dumazet	1	-3/+3
2019-01-29	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	50	-195/+605
2019-01-29	netfilter: nf_tables: add NFTA_RULE_POSITION_ID to nla_policy	Florian Westphal	1	-0/+1
2019-01-28	netfilter: conntrack: fix error path in nf_conntrack_pernet_init()	Cong Wang	1	-2/+2
2019-01-27	netrom: switch to sock timer API	Cong Wang	1	-10/+10
2019-01-22	net: introduce a knob to control whether to inherit devconf config	Cong Wang	5	-23/+58
2019-01-21	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	114	-602/+2090
2019-01-19	net_sched: add performance counters for basic filter	Cong Wang	2	-0/+32
2019-01-18	net/mlx5e: Force CHECKSUM_UNNECESSARY for short ethernet frames	Cong Wang	1	-0/+13
2019-01-18	net_sched: add hit counter for matchall	Cong Wang	2	-0/+30
2019-01-16	net/core/neighbour: fix kmemleak minimal reference count for hash tables	Konstantin Khlebnikov	1	-1/+1
2019-01-16	net_sched: refetch skb protocol for each filter	Cong Wang	1	-2/+1
2019-01-15	net/sched: act_tunnel_key: fix memory leak in case of action replace	Davide Caratti	1	-8/+11
2019-01-15	Revert "igb: reduce CPU0 latency when updating statistics"	Jeff Kirsher	3	-10/+10
2019-01-07	smc: move unhash as early as possible in smc_release()	Cong Wang	1	-1/+3
2019-01-03	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	60	-385/+2060
2019-01-01	ptr_ring: wrap back ->producer in __ptr_ring_swap_queue()	Cong Wang	1	-0/+2
2018-12-30	netrom: fix locking in nr_find_socket()	Cong Wang	1	-5/+10
2018-12-30	net/wan: fix a double free in x25_asy_open_tty()	Cong Wang	1	-0/+2
2018-12-30	ax25: fix a use-after-free in ax25_fillin_cb()	Cong Wang	2	-2/+11
2018-12-27	tipc: fix a double free in tipc_enable_bearer()	Cong Wang	1	-1/+0
2018-12-19	ipv6: explicitly initialize udp6_addr in udp_sock_create6()	Cong Wang	1	-1/+2
2018-12-18	tipc: check group dests after tipc_wait_for_cond()	Cong Wang	1	-4/+5
2018-12-15	net: sched: simplify the qdisc_leaf code	Tonghao Zhang	1	-3/+1
2018-12-14	tipc: check tsk->group in tipc_wait_for_cond()	Cong Wang	1	-11/+14
2018-12-14	net_sched: fold tcf_block_cb_call() into tc_setup_cb_call()	Cong Wang	6	-47/+35
2018-12-14	tipc: compare remote and local protocols in tipc_udp_enable()	Cong Wang	1	-0/+5
2018-12-14	tipc: fix a double kfree_skb()	Cong Wang	1	-3/+1
2018-12-14	tipc: use lock_sock() in tipc_sk_reinit()	Cong Wang	1	-2/+6
2018-12-01	net/sched: Replace call_rcu_bh() and rcu_barrier_bh()	Paul E. McKenney	2	-5/+5
2018-11-30	mlx5: fix get_ip_proto()	Cong Wang	1	-3/+3
2018-11-30	net/sched: act_police: fix memory leak in case of invalid control action	Davide Caratti	1	-12/+12
2018-11-29	net: explain __skb_checksum_complete() with comments	Cong Wang	2	-1/+18
2018-11-23	net/sched: act_police: add missing spinlock initialization	Davide Caratti	1	-0/+1
2018-11-16	net/sched: act_pedit: fix memory leak when IDR allocation fails	Davide Caratti	1	-1/+2
2018-11-16	net: remove unused skb_send_sock()	Cong Wang	2	-14/+0
2018-11-15	net: get rid of __tcp_checksum_complete()	Cong Wang	1	-6/+1
2018-11-15	net: dump more useful information in netdev_rx_csum_fault()	Cong Wang	5	-8/+16
2018-11-08	net: move __skb_checksum_complete*() to skbuff.c	Cong Wang	2	-43/+43
2018-11-01	net: drop skb on failure in ip_check_defrag()	Cong Wang	1	-4/+8
2018-10-31	net/mlx5e: fix csum adjustments caused by RXFCS	Eric Dumazet	1	-36/+9
2018-10-22	net/sched: act_police: disallow 'goto chain' on fallback control action	Davide Caratti	1	-2/+10
2018-10-22	net/sched: act_gact: disallow 'goto chain' on fallback control action	Davide Caratti	1	-0/+5
2018-10-19	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Greg Kroah-Hartman	57	-187/+253
2018-10-15	llc: set SOCK_RCU_FREE in llc_sap_add_socket()	Cong Wang	1	-0/+1
2018-10-15	net/sched: cls_api: add missing validation of netlink attributes	Davide Caratti	2	-9/+12
2018-10-05	net_sched: convert idrinfo->lock from spinlock to a mutex	Cong Wang	3	-35/+26
2018-10-01	bond: take rcu lock in netpoll_send_skb_on_dev	Dave Jones	1	-0/+2
2018-10-01	net_sched: fix a crash in tc_new_tfilter()	Cong Wang	1	-1/+3
2018-09-25	Merge branch 'Refactor-classifier-API-to-work-with-Qdisc-blocks-without-rtnl-...	David S. Miller	24	-112/+294
2018-09-24	ip_tunnel: be careful when accessing the inner header	Paolo Abeni	1	-0/+9
2018-09-21	net_sched: change tcf_del_walker() to take idrinfo->lock	Vlad Buslov	2	-3/+30
2018-09-13	ucma: fix a use-after-free in ucma_resolve_ip()	Cong Wang	1	-0/+2
2018-09-13	net_sched: notify filter deletion when deleting a chain	Cong Wang	1	-0/+2
2018-09-13	llc: avoid blocking in llc_sap_close()	Cong Wang	2	-3/+2
2018-09-12	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	68	-400/+593
2018-09-12	tipc: check return value of __tipc_dump_start()	Cong Wang	1	-1/+4
2018-09-12	rds: fix two RCU related problems	Cong Wang	1	-1/+4
2018-09-11	Merge git://git.kernel.org/pub/scm/linux/kernel/git/pablo/nf	David S. Miller	19	-100/+181
2018-09-11	netfilter: xt_hashlimit: use s->file instead of s->private	Cong Wang	1	-9/+9
2018-09-10	htb: use anonymous union for simplicity	Cong Wang	1	-49/+49
2018-09-10	net_sched: remove redundant qdisc lock classes	Cong Wang	1	-7/+0
2018-09-07	net_sched: properly cancel netlink dump on failure	Cong Wang	1	-4/+8
2018-09-06	tipc: call start and done ops directly in __tipc_nl_compat_dumpit()	Cong Wang	3	-6/+14
2018-09-05	net/sched: fix memory leak in act_tunnel_key_init()	Davide Caratti	1	-6/+10
2018-09-05	tipc: orphan sock in tipc_release()	Cong Wang	1	-0/+1
2018-09-04	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	89	-452/+941
2018-09-04	net: sched: action_ife: take reference to meta module	Vlad Buslov	1	-1/+15
2018-09-04	act_ife: fix a potential use-after-free	Cong Wang	1	-1/+1
2018-09-03	net: sched: null actions array pointer before releasing action	Vlad Buslov	1	-1/+1
2018-08-31	net_sched: add missing tcf_lock for act_connmark	Cong Wang	1	-3/+8
2018-08-31	Revert "net: sched: act: add extack for lookup callback"	Cong Wang	18	-38/+22
2018-08-29	net/sched: act_pedit: fix dump of extended layered op	Davide Caratti	1	-5/+13
2018-08-29	tipc: switch to rhashtable iterator	Cong Wang	4	-26/+56
2018-08-29	tipc: fix a missing rhashtable_walk_exit()	Cong Wang	1	-0/+2
2018-08-27	Merge branch 'net-sched-fixes'	David S. Miller	1	-2/+2
2018-08-26	net: sched: Fix memory exposure from short TCA_U32_SEL	Kees Cook	1	-2/+8
2018-08-22	addrconf: reduce unnecessary atomic allocations	Cong Wang	1	-3/+3
2018-08-21	Merge branch 'net_sched-fixes'	David S. Miller	30	-290/+108
2018-08-21	act_ife: fix a potential deadlock	Cong Wang	1	-13/+21
2018-08-21	act_ife: move tcfa_lock down to where necessary	Cong Wang	1	-25/+13
2018-08-21	Revert "net: sched: act_ife: disable bh when taking ife_mod_lock"	Cong Wang	1	-10/+10
2018-08-21	net_sched: remove unused tcfa_capab	Cong Wang	1	-2/+0
2018-08-21	net_sched: remove list_head from tc_action	Cong Wang	14	-60/+43
2018-08-21	net_sched: remove unused tcf_idr_check()	Cong Wang	2	-21/+3
2018-08-21	net_sched: remove unused parameter for tcf_action_delete()	Cong Wang	1	-3/+2
2018-08-21	net_sched: remove unnecessary ops->delete()	Cong Wang	18	-146/+7
2018-08-21	net_sched: improve and refactor tcf_action_put_many()	Cong Wang	1	-16/+15
2018-08-19	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	60	-352/+430
2018-08-16	ila: make lockdep happy again	Cong Wang	2	-7/+21
2018-08-16	net: sched: act_ife: always release ife action on init error	Vlad Buslov	1	-6/+2
2018-08-16	cls_matchall: fix tcf_unbind_filter missing	Hangbin Liu	1	-0/+2
2018-08-13	net_sched: Fix missing res info when create new tc_index filter	Hangbin Liu	1	-0/+1
2018-08-13	net_sched: fix NULL pointer dereference when delete tcindex filter	Hangbin Liu	1	-5/+2
2018-08-13	l2tp: use sk_dst_check() to avoid race on sk->sk_dst_cache	Wei Wang	1	-1/+1
2018-08-09	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	26	-161/+176
2018-08-07	llc: use refcount_inc_not_zero() for llc_sap_find()	Cong Wang	2	-2/+7
2018-08-07	vsock: split dwork to avoid reinitializations	Cong Wang	3	-11/+11
2018-08-05	ipv6: fix double refcount of fib6_metrics	Cong Wang	1	-4/+0
2018-08-01	net: sched: fix notifications for action-held chains	Jiri Pirko	1	-28/+43
2018-08-01	net: sched: change name of zombie chain to "held_by_acts_only"	Jiri Pirko	1	-6/+6
2018-07-26	net: rollback orig value on failure of dev_qdisc_change_tx_queue_len	Tariq Toukan	1	-7/+10
2018-07-24	net/sched: add skbprio scheduler	Nishanth Devarajan	4	-0/+349
2018-07-24	net: remove redundant input checks in SIOCSIFTXQLEN case of dev_ifsioc	Tariq Toukan	1	-6/+1
2018-07-04	net: sched: act_pedit: fix possible memory leak in tcf_pedit_init()	Wei Yongjun	1	-9/+16
2018-07-04	Merge branch 'Handle-multiple-received-packets-at-each-stage'	David S. Miller	11	-16/+360
2018-06-30	net: use dev_change_tx_queue_len() for SIOCSIFTXQLEN	Cong Wang	1	-9/+2
2018-06-24	net_sched: remove unused htb drop_list	Cong Wang	1	-13/+0
2018-06-23	net_sched: remove a bogus warning in hfsc	Cong Wang	1	-2/+2
2018-06-22	cls_flower: fix use after free in flower S/W path	Paolo Abeni	1	-4/+17
2018-06-20	net/sched: act_ife: preserve the action control in case of error	Davide Caratti	1	-2/+1
2018-06-20	net/sched: act_ife: fix recursive lock and idr leak	Davide Caratti	1	-5/+4
2018-06-12	smc: convert to ->poll_mask	Cong Wang	1	-9/+3
2018-06-10	socket: close race condition between sock_close() and sockfs_setattr()	Cong Wang	1	-3/+15
2018-06-05	netdev-FAQ: clarify DaveM's position for stable backports	Cong Wang	1	-0/+9
2018-06-04	infiniband: fix a possible use-after-free bug	Cong Wang	1	-1/+5
2018-05-24	net_sched: switch to rcu_work	Cong Wang	13	-221/+85
2018-05-14	tun: fix use after free for ptr_ring	Jason Wang	1	-15/+12
2018-05-11	net sched actions: fix refcnt leak in skbmod	Roman Mashak	1	-1/+4
2018-05-11	net: sched: fix error path in tcf_proto_create() when modules are not configured	Jiri Pirko	1	-1/+1
2018-05-11	net sched actions: fix invalid pointer dereferencing if skbedit flags missing	Roman Mashak	1	-1/+2
2018-05-06	Merge git://git.kernel.org/pub/scm/linux/kernel/git/pablo/nf-next	David S. Miller	104	-1887/+2753
2018-04-27	ipvs: initialize tbl->entries in ip_vs_lblc_init_svc()	Cong Wang	1	-0/+1
2018-04-27	ipvs: initialize tbl->entries after allocation	Cong Wang	1	-0/+1
2018-04-23	Merge git://git.kernel.org/pub/scm/linux/kernel/git/pablo/nf	David S. Miller	10	-171/+200
2018-04-22	llc: fix NULL pointer deref for SOCK_ZAPPED	Cong Wang	1	-9/+12
2018-04-22	llc: delete timers synchronously in llc_sk_free()	Cong Wang	3	-9/+23
2018-04-20	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	69	-349/+786
2018-04-19	llc: hold llc_sap before release_sock()	Cong Wang	1	-0/+7
2018-04-17	VSOCK: make af_vsock.ko removable again	Stefan Hajnoczi	1	-0/+6
2018-04-16	netfilter: conntrack: silent a memory leak warning	Cong Wang	1	-0/+1
2018-04-11	proc: reject "." and ".." as filenames	Alexey Dobriyan	1	-0/+8
2018-04-09	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	53	-298/+553
2018-04-08	tipc: use the right skb in tipc_sk_fill_sock_diag()	Cong Wang	3	-6/+6
2018-04-07	net_sched: fix a missing idr_remove() in u32_delete_key()	Cong Wang	1	-0/+1
2018-04-04	af_unix: remove redundant lockdep class	Cong Wang	1	-10/+0
2018-03-30	Merge git://git.kernel.org/pub/scm/linux/kernel/git/pablo/nf-next	David S. Miller	75	-858/+1383
2018-03-27	llc: properly handle dev_queue_xmit() return value	Cong Wang	3	-16/+33
2018-03-23	net/sched: remove tcf_idr_cleanup()	Davide Caratti	2	-9/+0
2018-03-21	Merge branch 'net-sched-action-idr-leak'	David S. Miller	7	-9/+12
2018-03-21	net/sched: fix idr leak in the error path of tcf_simp_init()	Davide Caratti	1	-1/+1
2018-03-17	net: sched: fix uses after free	Eric Dumazet	1	-9/+13
2018-03-15	net sched actions: return explicit error when tunnel_key mode is not specified	Roman Mashak	1	-0/+1
2018-03-06	perf stat: Fix CVS output format for non-supported counters	Ilya Pronin	1	-1/+1
2018-03-05	netfilter: make xt_rateest hash table per net	Cong Wang	3	-33/+72
2018-02-23	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	108	-455/+625
2018-02-21	Merge git://git.kernel.org/pub/scm/linux/kernel/git/pablo/nf	David S. Miller	55	-300/+314
2018-02-20	net: sched: report if filter is too large to dump	Roman Kapl	1	-1/+6
2018-02-14	netfilter: ipt_CLUSTERIP: fix a refcount bug in clusterip_config_find_get()	Cong Wang	1	-2/+6
2018-02-13	net: sched: fix tc_u_common lookup	Jiri Pirko	1	-4/+20
2018-02-13	net: sched: don't set q pointer for shared blocks	Jiri Pirko	1	-14/+12
2018-02-08	netfilter: ipt_CLUSTERIP: fix a race condition of proc file creation	Cong Wang	1	-6/+6

[prev] [next]

index : kernel/git/netdev/net-next.git

Netdev Group's -next networking tree

main switch

Netdev Group

about summary refs log tree commit diff stats

log msg Cong Wang search

Age	Commit message (Expand)	Author	Files	Lines
2018-02-07	Merge git://git.kernel.org/pub/scm/linux/kernel/git/pablo/nf	David S. Miller	15	-79/+97
2018-02-07	netfilter: xt_RATEEST: acquire xt_rateest_mutex for hash insert	Cong Wang	1	-5/+17
2018-02-06	cls_u32: fix use after free in u32_destroy_key()	Paolo Abeni	1	-10/+11
2018-02-02	cls_u32: add missing RCU annotation.	Paolo Abeni	1	-5/+7
2018-02-02	netfilter: xt_cgroup: initialize info->priv in cgroup_mt_check_v1()	Cong Wang	1	-0/+1
2018-01-29	net_sched: gen_estimator: fix lockdep splat	Eric Dumazet	1	-0/+4
2018-01-29	Merge branch 'net_sched-reflect-tx_queue_len-change-for-pfifo_fast'	David S. Miller	6	-37/+89
2018-01-29	net_sched: implement ->change_tx_queue_len() for pfifo_fast	Cong Wang	1	-0/+18
2018-01-29	net_sched: plug in qdisc ops change_tx_queue_len	Cong Wang	3	-0/+36
2018-01-29	net: introduce helper dev_change_tx_queue_len()	Cong Wang	4	-37/+35
2018-01-22	tun: avoid calling xdp_rxq_info_unreg() twice	Cong Wang	1	-2/+0
2018-01-19	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	David S. Miller	115	-751/+1723
2018-01-19	Merge branch 'net-sched-cls-add-extack-support'	David S. Miller	14	-92/+194
2018-01-19	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	27	-117/+369
2018-01-17	tun: fix a memory leak for tfile->tx_array	Cong Wang	1	-2/+13
2018-01-16	printk: Never set console_may_schedule in console_trylock()	Sergey Senozhatsky	1	-14/+8
2018-01-16	printk: Hide console waiter logic into helpers	Petr Mladek	1	-97/+148
2018-01-16	printk: Add console owner and waiter logic to load balance console writes	Steven Rostedt (VMware)	1	-1/+107
2018-01-15	tipc: fix a memory leak in tipc_nl_node_get_link()	Cong Wang	1	-12/+14
2018-01-10	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	37	-131/+303
2018-01-10	8021q: fix a memory leak for VLAN 0 device	Cong Wang	1	-6/+1
2018-01-05	net: sched: fix tcf_block_get_ext() in case CONFIG_NET_CLS is not set	Quentin Monnet	1	-1/+2
2017-12-28	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	30	-121/+298
2017-12-27	Merge branch 'master' of git://git.kernel.org/pub/scm/linux/kernel/git/klasse...	David S. Miller	7	-5/+125
2017-12-26	net_sched: fix a missing rcu barrier in mini_qdisc_pair_swap()	Cong Wang	1	-1/+3
2017-12-19	net_sched: properly check for empty skb array on error path	Cong Wang	1	-1/+7
2017-12-15	net: sched: fix static key imbalance in case of ingress/clsact_init error	Jiri Pirko	1	-4/+5
2017-12-15	net: sched: fix clsact init error path	Jiri Pirko	2	-7/+3
2017-12-13	net_sched: switch to exit_batch for action pernet ops	Cong Wang	17	-88/+61
2017-12-11	netlink: convert netlink tap spinlock to mutex	Cong Wang	1	-6/+6
2017-12-11	netlink: make netlink tap per netns	Cong Wang	1	-17/+49
2017-12-06	act_mirred: get rid of mirred_list_lock spinlock	Cong Wang	1	-9/+1
2017-12-06	act_mirred: get rid of tcfm_ifindex from struct tcf_mirred	Cong Wang	11	-41/+26
2017-12-05	net_sched: remove unused parameter from act cleanup ops	Cong Wang	12	-16/+16
2017-12-05	net_sched: get rid of rcu_barrier() in tcf_block_put_ext()	Cong Wang	2	-22/+9
2017-12-05	tipc: fix a null pointer deref on error path	Cong Wang	1	-1/+1
2017-11-30	act_sample: get rid of tcf_sample_cleanup_rcu()	Cong Wang	2	-12/+3
2017-11-29	xfrm: check id proto in validate_tmpl()	Cong Wang	1	-0/+15
2017-11-28	sch_sfq: fix null pointer dereference at timer expiration	Paolo Abeni	1	-0/+1
2017-11-28	cls_bpf: don't decrement net's refcount when offload fails	Jakub Kicinski	1	-10/+13
2017-11-15	mm: don't warn about allocations which stall for too long	Tetsuo Handa	1	-10/+0
2017-11-11	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	15	-34/+68
2017-11-11	vlan: fix a use-after-free in vlan_device_event()	Cong Wang	1	-3/+3
2017-11-09	Merge branch 'net-sched-race-fix'	David S. Miller	31	-63/+198
2017-11-09	cls_u32: use tcf_exts_get_net() before call_rcu()	Cong Wang	1	-1/+7
2017-11-09	cls_tcindex: use tcf_exts_get_net() before call_rcu()	Cong Wang	1	-7/+25
2017-11-09	cls_rsvp: use tcf_exts_get_net() before call_rcu()	Cong Wang	1	-3/+12
2017-11-09	cls_route: use tcf_exts_get_net() before call_rcu()	Cong Wang	1	-3/+14
2017-11-09	cls_matchall: use tcf_exts_get_net() before call_rcu()	Cong Wang	1	-3/+12
2017-11-09	cls_fw: use tcf_exts_get_net() before call_rcu()	Cong Wang	1	-3/+14
2017-11-09	cls_flower: use tcf_exts_get_net() before call_rcu()	Cong Wang	1	-3/+13
2017-11-09	cls_flow: use tcf_exts_get_net() before call_rcu()	Cong Wang	1	-6/+18
2017-11-09	cls_cgroup: use tcf_exts_get_net() before call_rcu()	Cong Wang	1	-6/+18
2017-11-09	cls_bpf: use tcf_exts_get_net() before call_rcu()	Cong Wang	1	-1/+6
2017-11-09	cls_basic: use tcf_exts_get_net() before call_rcu()	Cong Wang	1	-5/+15
2017-11-09	net_sched: introduce tcf_exts_get_net() and tcf_exts_put_net()	Cong Wang	2	-0/+25
2017-11-09	Revert "net_sched: hold netns refcnt for each action"	Cong Wang	18	-22/+18
2017-11-03	net_sched: check NULL in tcf_block_put()	Cong Wang	1	-3/+2
2017-11-03	Merge branch 'net-sched-use-after-free'	David S. Miller	18	-18/+26
2017-11-03	net_sched: hold netns refcnt for each action	Cong Wang	18	-18/+22
2017-11-03	net_sched: acquire RTNL in tc_action_net_exit()	Cong Wang	2	-0/+4
2017-11-02	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	David S. Miller	47	-169/+284
2017-11-01	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	26	-86/+182
2017-10-31	net_sched: remove tcf_block_put_deferred()	Cong Wang	1	-29/+8
2017-10-29	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	78	-361/+1021
2017-10-29	Merge branch 'net_sched-fix-races-with-RCU-callbacks'	David S. Miller	19	-57/+367
2017-10-29	selftests: Introduce a new test case to tc testsuite	Chris Mi	3	-5/+40
2017-10-29	selftests: Introduce a new script to generate tc batch file	Chris Mi	1	-0/+62
2017-10-29	net_sched: fix call_rcu() race on act_sample module removal	Cong Wang	1	-0/+1
2017-10-29	net_sched: add rtnl assertion to tcf_exts_destroy()	Cong Wang	1	-0/+1
2017-10-29	net_sched: use tcf_queue_work() in tcindex filter	Cong Wang	1	-5/+33
2017-10-29	net_sched: use tcf_queue_work() in rsvp filter	Cong Wang	1	-3/+16
2017-10-29	net_sched: use tcf_queue_work() in route filter	Cong Wang	1	-3/+16
2017-10-29	net_sched: use tcf_queue_work() in u32 filter	Cong Wang	1	-3/+26
2017-10-29	net_sched: use tcf_queue_work() in matchall filter	Cong Wang	1	-3/+16
2017-10-29	net_sched: use tcf_queue_work() in fw filter	Cong Wang	1	-3/+16
2017-10-29	net_sched: use tcf_queue_work() in flower filter	Cong Wang	1	-3/+16
2017-10-29	net_sched: use tcf_queue_work() in flow filter	Cong Wang	1	-3/+16
2017-10-29	net_sched: use tcf_queue_work() in cgroup filter	Cong Wang	1	-4/+18
2017-10-29	net_sched: use tcf_queue_work() in bpf filter	Cong Wang	1	-2/+17
2017-10-29	net_sched: use tcf_queue_work() in basic filter	Cong Wang	1	-3/+17
2017-10-29	net_sched: introduce a workqueue for RCU callbacks of tc filter	Cong Wang	3	-17/+56
2017-10-29	net_sched: avoid matching qdisc with zero handle	Cong Wang	1	-0/+2
2017-10-26	tipc: fix a dangling pointer	Cong Wang	1	-1/+3
2017-10-26	vsock: always call vsock_init_tables()	Cong Wang	1	-3/+4
2017-10-26	tun: allow positive return values on dev_get_valid_name() call	Julien Gomes	1	-1/+1
2017-10-22	Merge branch 'tun-timer-cleanups'	David S. Miller	1	-7/+8
2017-10-18	tcp: Check daddr_cache before use in tracepoint	David Ahern	1	-4/+4
2017-10-18	tcp: Use pI6c in tcp tracepoint	David Ahern	1	-1/+1
2017-10-18	net: sched: Convert timers to use timer_setup()	Kees Cook	5	-20/+23
2017-10-16	tun: call dev_get_valid_name() before register_netdevice()	Cong Wang	3	-3/+9
2017-10-14	tcp: add a tracepoint for tcp retransmission	Cong Wang	3	-0/+72
2017-10-14	net_sched: fix a compile warning in act_ife	Cong Wang	1	-0/+2
2017-10-12	net sched actions: fix module auto-loading	Roman Mashak	1	-1/+15
2017-10-01	net_sched: remove redundant assignment to ret	Colin Ian King	1	-1/+0
2017-09-28	net_sched: use idr to allocate u32 filter handles	Cong Wang	1	-41/+67
2017-09-28	net_sched: use idr to allocate basic filter handles	Cong Wang	1	-15/+22
2017-09-28	net_sched: use idr to allocate bpf filter handles	Cong Wang	1	-29/+28
2017-09-21	net_sched: remove cls_flower idr on failure	Cong Wang	1	-6/+9
2017-09-19	net_sched: no need to free qdisc in RCU callback	Cong Wang	2	-9/+2
2017-09-12	Merge branch 'net_sched-fix-filter-chain-reference-counting'	David S. Miller	3	-34/+48
2017-09-12	net_sched: carefully handle tcf_block_put()	Cong Wang	1	-6/+16
2017-09-12	net_sched: fix reference counting of tc filter chain	Cong Wang	1	-21/+26
2017-09-12	net_sched: get rid of tcfa_rcu	Cong Wang	2	-11/+8
2017-09-11	net/sched: fix pointer check in gen_handle	Josh Hunt	1	-1/+1
2017-08-31	wl1251: add a missing spin_lock_init()	Cong Wang	1	-0/+1
2017-08-31	net_sched: add reverse binding for tc class	Cong Wang	11	-2/+148
2017-08-25	Merge branch 'net_sched-clean-up-tc-classes-and-u32-filter'	David S. Miller	22	-452/+361
2017-08-25	net_sched: kill u32_node pointer in Qdisc	WANG Cong	2	-5/+53
2017-08-25	net_sched: remove tc class reference counting	WANG Cong	21	-262/+106
2017-08-25	net_sched: introduce tclass_del_notify()	WANG Cong	1	-6/+33
2017-08-25	net_sched: get rid of more forward declarations	WANG Cong	1	-179/+169
2017-08-20	net_sched: fix order of queue length updates in qdisc_replace()	Konstantin Khlebnikov	1	-1/+4
2017-08-18	net_sched: fix p_filter_chain check in tcf_chain_flush	Jiri Pirko	1	-1/+1
2017-08-11	net/sched/hfsc: allocate tcf block for hfsc root class	Konstantin Khlebnikov	1	-0/+6
2017-08-10	mm: fix list corruptions on shmem shrinklist	Cong Wang	1	-2/+10
2017-08-09	net: sched: set xt_tgchk_param par.nft_compat as 0 in ipt_init_target	Xin Long	1	-1/+1
2017-08-08	net_sched: get rid of some forward declarations	WANG Cong	1	-111/+103
2017-08-07	Merge branch 'net_sched-clean-up-filter-handle'	David S. Miller	14	-154/+178
2017-08-07	net_sched: use void pointer for filter handle	WANG Cong	14	-151/+141
2017-08-07	net_sched: refactor notification code for RTM_DELTFILTER	WANG Cong	1	-5/+39
2017-08-02	flow_dissector: remove unused functions	WANG Cong	2	-61/+0
2017-07-31	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	80	-296/+629
2017-07-29	team: use a larger struct for mac address	WANG Cong	1	-4/+4
2017-07-29	net: check dev->addr_len for dev_set_mac_address()	WANG Cong	1	-0/+2
2017-07-27	wl1251: add a missing spin_lock_init()	Cong Wang	1	-0/+1
2017-07-26	bonding: commit link status change after propose	WANG Cong	1	-0/+2
2017-07-24	packet: fix use-after-free in prb_retire_rx_blk_timer_expired()	WANG Cong	1	-1/+1
2017-07-20	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	83	-581/+1176
2017-07-20	rtnetlink: allocate more memory for dev_set_mac_address()	WANG Cong	1	-1/+2
2017-07-17	bpf: check NULL for sk_to_full_sk() return value	WANG Cong	1	-1/+1
2017-07-14	netpoll: shut up a kernel warning on refcount	WANG Cong	1	-1/+1
2017-07-12	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	38	-101/+172
2017-07-11	tap: convert a mutex to a spinlock	WANG Cong	1	-9/+9
2017-07-09	mqueue: fix a use-after-free in sys_mq_notify()	Cong Wang	1	-1/+3
2017-07-08	bonding: avoid NETDEV_CHANGEMTU event when unregistering slave	WANG Cong	3	-7/+12
2017-06-25	tcp: reset sk_rx_dst in tcp_disconnect()	WANG Cong	1	-0/+2
2017-06-23	sit: use __GFP_NOWARN for user controlled allocation	WANG Cong	1	-1/+1
2017-06-22	ipv6: avoid unregistering inet6_dev for loopback	WANG Cong	1	-2/+3
2017-06-22	ipv6: only call ip6_route_dev_notify() once for NETDEV_UNREGISTER	WANG Cong	1	-1/+5
2017-06-20	igmp: add a missing spin_lock_init()	WANG Cong	1	-0/+1
2017-06-14	net_sched: move tcf_lock down after gen_replace_estimator()	WANG Cong	1	-5/+3
2017-06-13	igmp: acquire pmc lock for ip_mc_clear_src()	WANG Cong	1	-8/+13
2017-06-10	sctp: fix recursive locking warning in sctp_do_peeloff	Xin Long	1	-2/+2
2017-06-06	e1000e: use disable_hardirq() also for MSIX vectors in e1000_netpoll()	Konstantin Khlebnikov	1	-6/+6
2017-05-26	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	57	-254/+701
2017-05-26	ipv4: add reference counting to metrics	Eric Dumazet	5	-23/+45
2017-05-25	net_sched: only create filter chains for new filters/actions	WANG Cong	3	-6/+12
2017-05-23	net/sched: fix filter flushing	Jiri Pirko	1	-3/+10
2017-05-22	vsock: use new wait API for vsock_stream_sendmsg()	WANG Cong	1	-13/+8
2017-05-11	net: sched: optimize class dumps	Eric Dumazet	1	-0/+6
2017-05-11	ipv6/dccp: do not inherit ipv6_mc_list from parent	WANG Cong	2	-0/+8
2017-05-08	ipv6: reorder ip6_route_dev_notifier after ipv6_dev_notf	WANG Cong	3	-2/+14
2017-05-08	ipv4: restore rt->fi for reference counting	WANG Cong	2	-1/+18
2017-05-05	x86/asm: Don't use RBP as a temporary register in csum_partial_copy_generic()	Josh Poimboeuf	1	-6/+6
2017-05-04	ipv6: initialize route null entry in addrconf_init()	WANG Cong	3	-11/+18
2017-05-02	audit: fix the RCU locking for the auditd_connection structure	Paul Moore	1	-57/+100
2017-04-30	ipv4: get rid of ip_ra_lock	WANG Cong	1	-16/+2
2017-04-26	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	44	-141/+373
2017-04-26	ipv6: check skb->protocol before lookup for nexthop	WANG Cong	1	-16/+18
2017-04-25	ipv6: move stub initialization after ipv6 setup completion	Paolo Abeni	1	-2/+4
2017-04-24	cls_flower: add support for matching MPLS fields (v2)	Benjamin LaHaise	2	-0/+79
2017-04-24	flow_dissector: add mpls support (v2)	Benjamin LaHaise	3	-3/+35
2017-04-21	Merge branch 'tc-filter-cleanup-destroy-delete'	David S. Miller	13	-148/+134
2017-04-21	net_sched: remove useless NULL to tp->root	WANG Cong	3	-20/+0
2017-04-21	net_sched: move the empty tp check from ->destroy() to ->delete()	WANG Cong	13	-128/+134
2017-04-20	Add Cong Wang as TC subsystem co-maintainer	Jamal Hadi Salim	1	-0/+1
2017-04-19	nsfs: mark dentry with DCACHE_RCUACCESS	Cong Wang	1	-0/+1
2017-04-18	Merge git://git.kernel.org/pub/scm/linux/kernel/git/davem/net	Linus Torvalds	15	-58/+69
2017-04-17	kcm: remove a useless copy_from_user()	WANG Cong	1	-4/+0
2017-04-17	ipv4: fix a deadlock in ip_ra_control	WANG Cong	3	-9/+5
2017-04-06	net_sched: check noop_qdisc before qdisc_hash_add()	WANG Cong	1	-1/+1
2017-03-24	kcm: return immediately after copy_from_user() failure	WANG Cong	1	-3/+3
2017-03-16	bridge: resolve a false alarm of lockdep	WANG Cong	2	-10/+1
2017-03-16	Merge branch 'sched-cleanups'	David S. Miller	2	-4/+2
2017-03-16	net/sched: fq_codel: Avoid set-but-unused variable	Or Gerlitz	1	-2/+0
2017-03-16	ipvs: remove an annoying printk in netns init	Cong Wang	1	-2/+0
2017-03-12	act_connmark: avoid crashing on malformed nlattrs with null parms	Etienne Noss	1	-0/+3
2017-03-09	net/socket: use per af lockdep classes for sk queues	Paolo Abeni	1	-18/+78
2017-03-08	team: use ETH_MAX_MTU as max mtu	Jarod Wilson	1	-0/+1
2017-03-07	ipv6: reorder icmpv6_init() and ip6_mr_init()	WANG Cong	1	-5/+5
2017-03-03	strparser: destroy workqueue on module exit	WANG Cong	1	-0/+1
2017-03-02	bonding: use ETH_MAX_MTU as max mtu	WANG Cong	1	-0/+1
2017-03-02	ipv6: ignore null_entry in inet6_rtm_getroute() too	WANG Cong	1	-0/+6
2017-03-02	net: Introduce sk_clone_lock() error path routine	Arnaldo Carvalho de Melo	3	-10/+13
2017-03-02	dccp: Unlock sock before calling sk_free()	Arnaldo Carvalho de Melo	1	-0/+1
2017-03-02	ipv6: check for ip6_null_entry in __ip6_del_rt_siblings()	WANG Cong	1	-5/+9
2017-02-26	net sched actions: decrement module reference count after table flush.	Roman Mashak	1	-4/+1
2017-02-22	9p: fix a potential acl leak	Cong Wang	1	-0/+2
2017-02-14	kcm: fix a null pointer dereference in kcm_sendmsg()	WANG Cong	1	-2/+4
2017-02-09	kcm: fix 0-length case for kcm_sendmsg()	WANG Cong	1	-18/+22
2017-02-08	ping: fix a null pointer dereference	WANG Cong	1	-0/+2
2017-02-08	sit: fix a double free on error path	WANG Cong	1	-0/+1
2017-02-03	net: use a work queue to defer net_disable_timestamp() work	Eric Dumazet	1	-18/+13
2017-02-01	Merge branch 'act_sample-Little-fixes'	David S. Miller	1	-2/+4
2017-02-01	net/sched: act_psample: Remove unnecessary ASSERT_RTNL	Yotam Gigi	1	-1/+0
2017-02-01	net/sched: act_sample: Fix error path in init	Yotam Gigi	1	-1/+4
2017-01-24	af_unix: move unix_mknod() out of bindlock	WANG Cong	1	-11/+16
2016-12-26	net_sched: fix soft lockup in tc_classify	Daniel Borkmann	1	-1/+3
2016-12-14	audit: use proper refcount locking on audit_sock	Richard Guy Briggs	1	-6/+24

[prev] [next]

Actually, a huge amount of kernel code

If Cong Wang doesn’t know what this thing is….

then it’s *not too surprising* I had no idea what it did either.

Some documentation in the man page (man 7 socket)

SO_INCOMING_NAPI_ID (gettable since Linux 4.12)

Returns a system-level unique ID called NAPI ID that is associated with a RX queue on which the last packet associated with that socket is received.

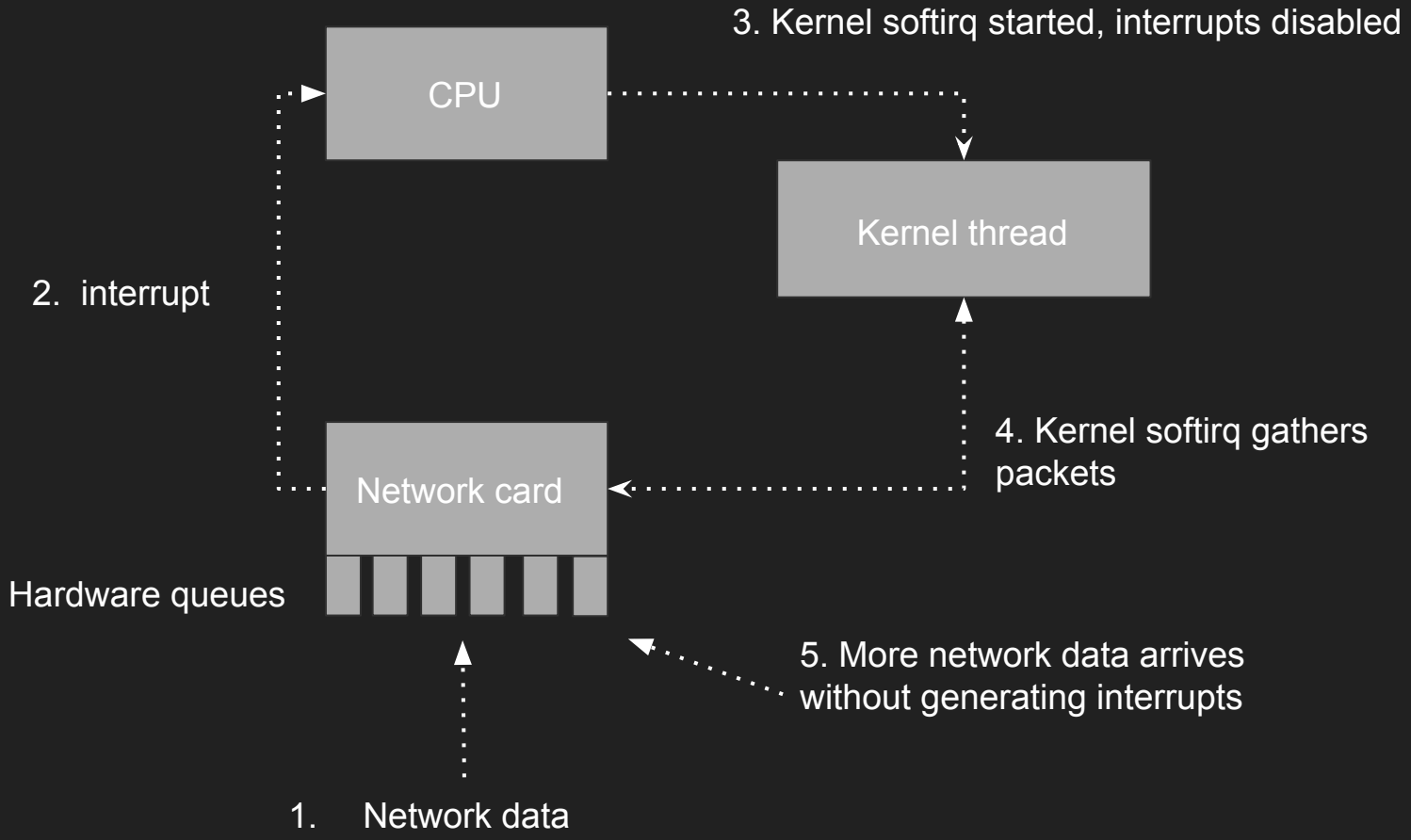
This can be used by an application to split the incoming flows among worker threads based on the RX queue on which the packets associated with the flows are received. It allows each worker thread to be associated with a NIC HW receive queue and service all the connection requests received on that RX queue. This mapping between a app thread and a HW NIC queue streamlines the flow of data from the NIC to the application.

Vaguely seemed like flow steering?

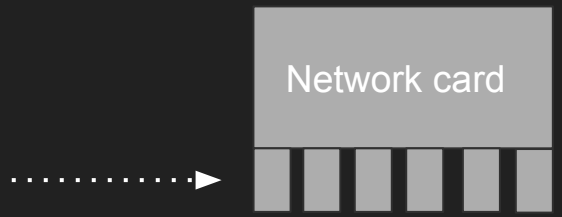
Supported by memcached

Introduce NAPI ID based worker thread selection

```
By default memcached assigns connections to worker threads in
a round-robin manner. This patch introduces an option to select
a worker thread based on the incoming connection's NAPI ID if
SO_INCOMING_NAPI_ID socket option is supported by the OS.

This allows a memcached worker thread to be associated with a
NIC HW receive queue and service all the connection requests
received on a specific RX queue. This mapping between a memcached
thread and a HW NIC queue streamlines the flow of data from the
NIC to the application. In addition, an optimal path with reduced
context switches is possible, if epoll based busy polling
(sysctl -w net.core.busy_poll = <non-zero value>) is also enabled.

This feature is enabled via a new command line parameter -N <num>
or "--napi_ids=<num>", where <num> is the number of available/assigned
NIC hardware RX queues through which the connections can be received.
The number of napi_ids specified cannot be greater than the number
of worker threads specified using -t/--threads option.
If the option is not specified, or the conditions not met, the code
defaults to round robin thread selection.

Signed-off-by: Kiran Patil <kiran.patil@intel.com>
Signed-off-by: Sridhar Samudrala <sridhar.samudrala@intel.com>
```


3. Kernel softirq started, interrupts disabled
CPU
Kernel thread
2. interrupt
4. Kernel softirq gathers packets
Network card
Hardware queues
5. More network data arrives without generating interrupts
1. Network data

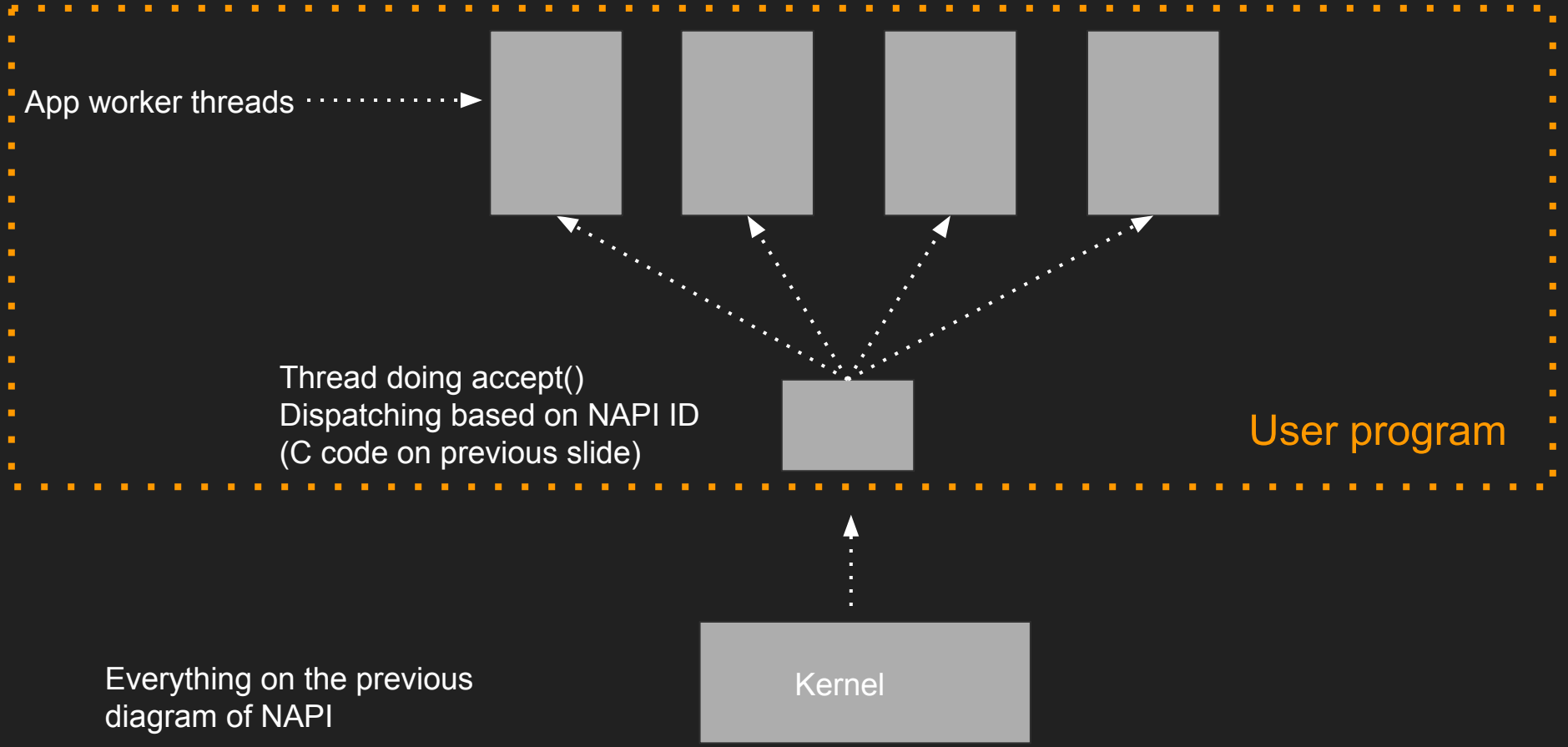
Hardware queues, each with their own system-wide unique ID (aka NAPI ID)



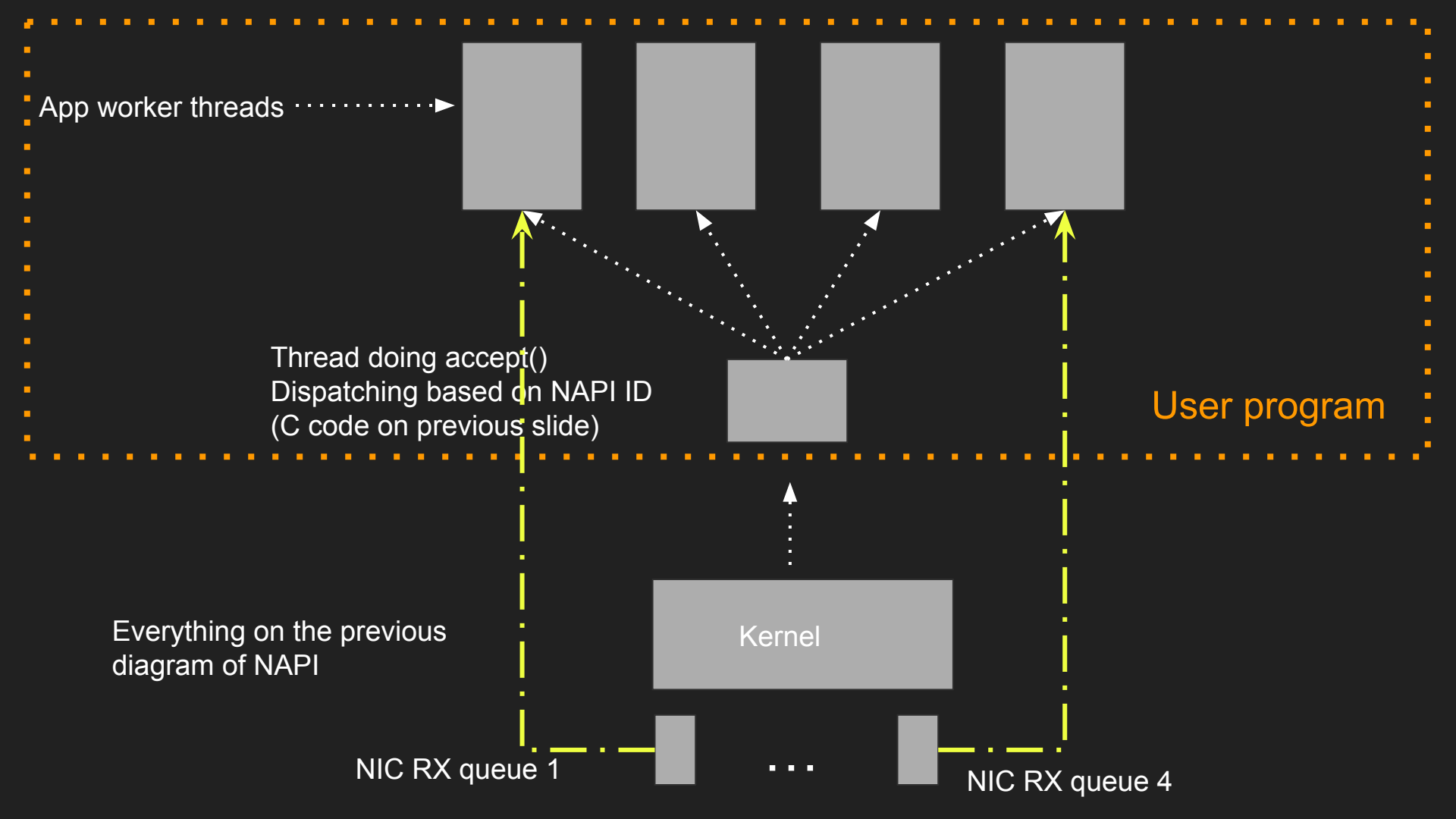
1. Network data

Example (pseudo code) usage

```
fd = accept(server_fd);
getsockopt(fd, SOL_SOCKET, SO_INCOMING_NAPI_ID, &napi_id, &napi_id_len);

/* now napi_id tells you which hardware queue this connection arrived on */
dispatch_to_thread_by_napi_id(fd, napi_id);
```



Everything on the previous diagram of NAPI

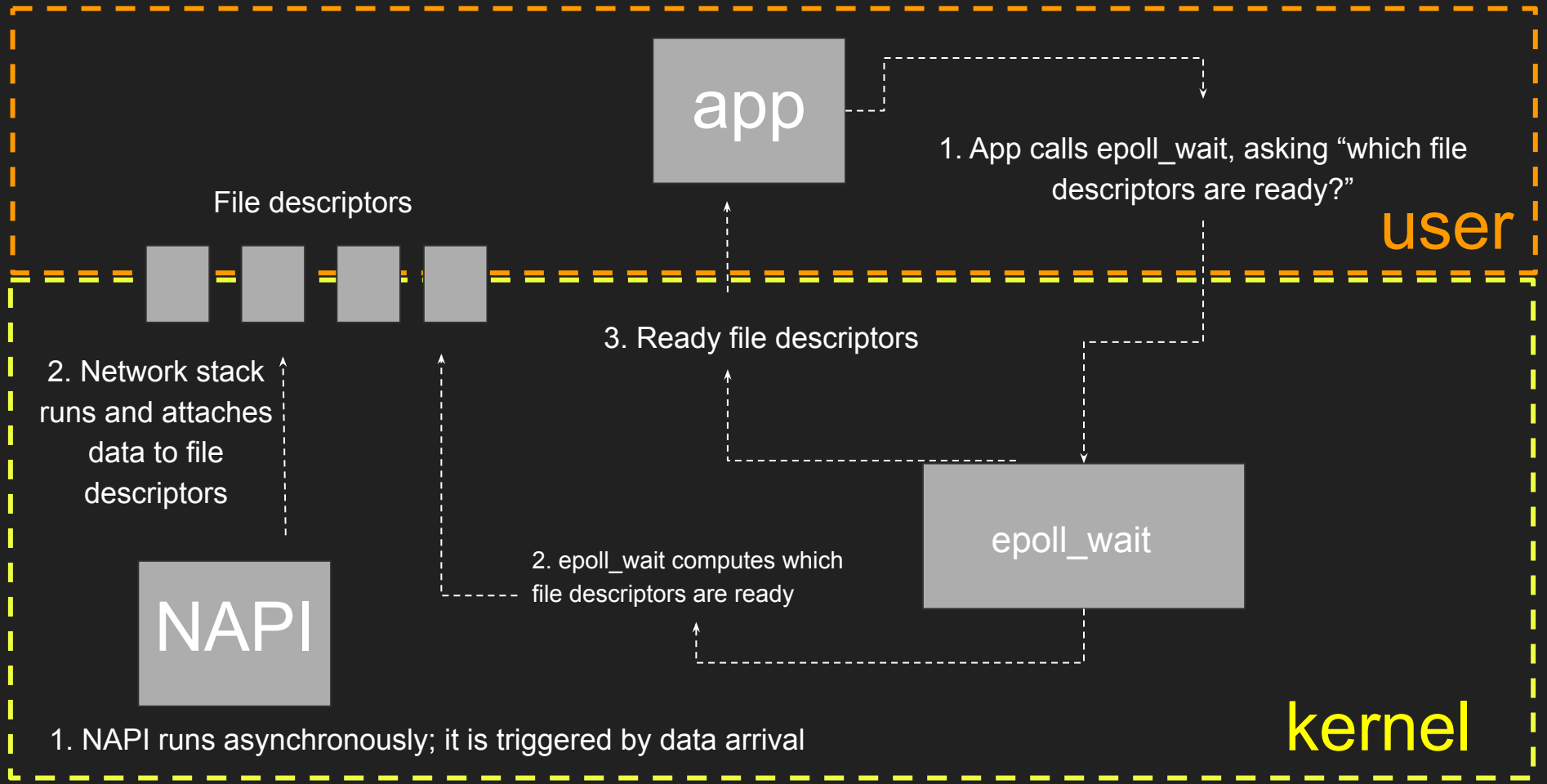

App worker threads
Thread doing accept()
Dispatching based on NAPI ID
(C code on previous slide)
User program
Everything on the previous
diagram of NAPI
Kernel
NIC RX queue 1
NIC RX queue 4

SO_INCOMING_NAPI_ID

- Provides hardware queue ID associated with connection
- Application can select worker thread based on queue ID
- Allows apps to map hardware NIC queues to threads
- Should improve cache hit rates and memory locality

But… there’s more.

Before we can proceed, let’s look at how epoll_wait + NAPI work normally at a high level



We’ll assume this all happens on the same CPU

That’s how “normal” epoll_wait works, but…

An epoll specific modification was added to the kernel for SO_INCOMING_NAPI_ID

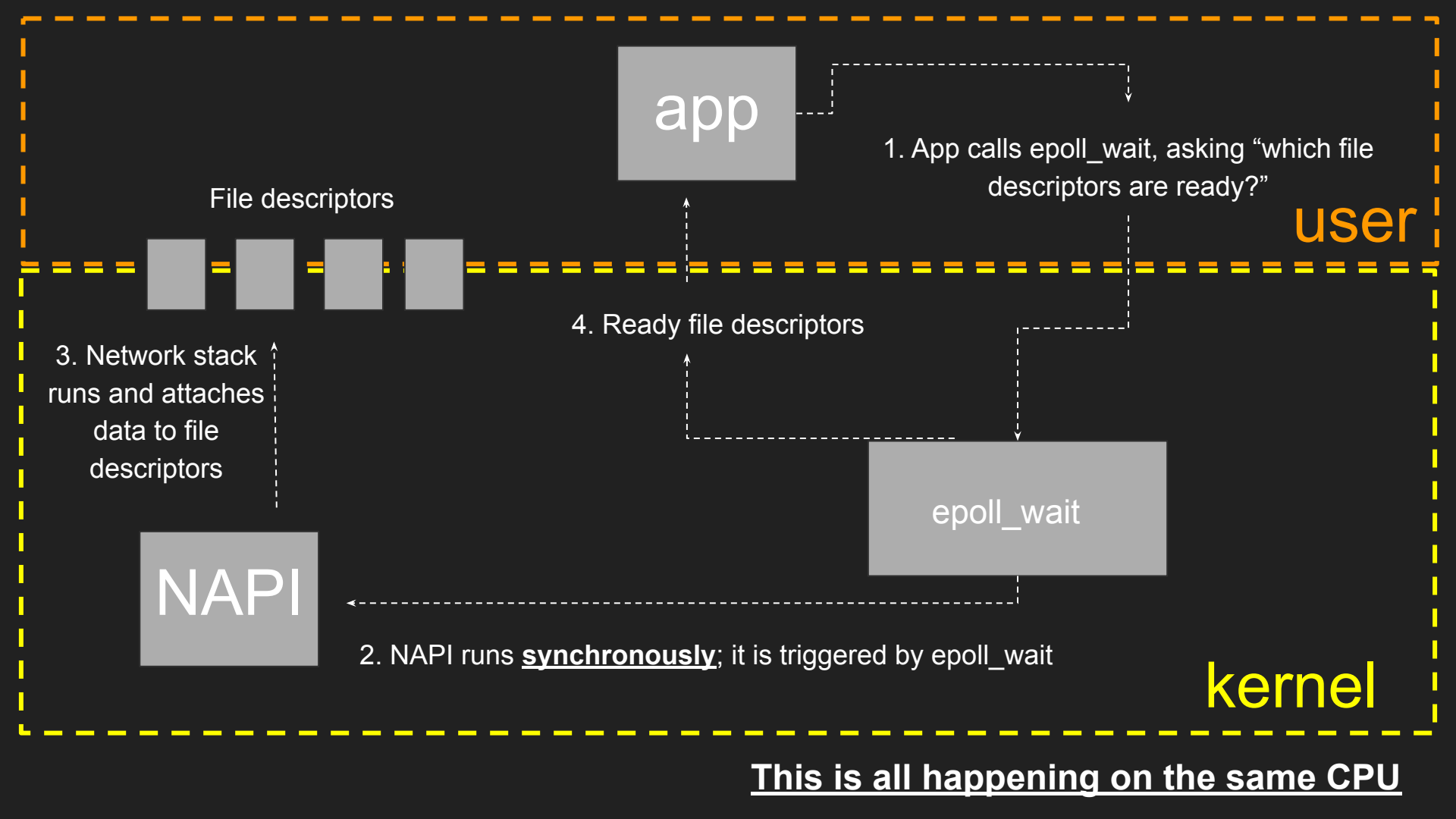
The SO_INCOMING_NAPI_ID of the last fd added to the epoll set can be busy polled.

To do so, you'd want to ensure all fds added are from the same RX queue

Set a *system-wide* sysctl or proc value

/proc/sys/net/core/busy_poll

Kind of looks like this


app
1. App calls epoll_wait, asking “which file descriptors are ready?”
user
File descriptors
4. Ready file descriptors
3. Network stack runs and attaches data to file descriptors
epoll_wait
NAPI
2. NAPI runs synchronously; it is triggered by epoll_wait
kernel
This is all happening on the same CPU

epoll + SO_INCOMING_NAPI_ID

1. All FDs must have same NAPI_ID
2. epoll_wait drives packet processing
3. Probably better cache efficiency
4. The app can decide to do network processing vs its own work
5. *But this is limited by device IRQs which will still fire*

But it’s system wide… so this makes everything suddenly busy poll

So…. I wrote a patch to make it per epoll-context specific via an ioctl

@ 2024-02-13 6:16 Joe Damato
  2024-02-13 6:16 ` [PATCH net-next v8 1/4] eventpoll: support busy poll per epoll instance Joe Damato
                   ` (4 more replies)
  0 siblings, 5 replies; 7+ messages in thread
From: Joe Damato @ 2024-02-13 6:16 UTC (permalink / raw)
  To: linux-kernel, netdev
  Cc: chuck.lever, jlayton, linux-api, brauner, edumazet, davem,
      alexander.duyck, sridhar.samudrala, kuba, willemdebruijn.kernel,
      weiwan, David.Laight, arnd, sdf, amritha.nambiar, Joe Damato,
      Alexander Viro, Greg Kroah-Hartman, Helge Deller, Jan Kara,
      Jiri Slaby, Jonathan Corbet, Julien Panis,
      open list:DOCUMENTATION,
      open list:FILESYSTEMS (VFS and infrastructure),
      Michael Ellerman, Nathan Lynch, Palmer Dabbelt, Steve French,
      Thomas Huth, Thomas Zimmermann

Greetings:

Welcome to v8.

TL;DR This builds on commit bf3b9f6372c4 ("epoll: Add busy poll support to
epoll with socket fds.") by allowing user applications to enable
epoll-based busy polling, set a busy poll packet budget, and enable or
disable prefer busy poll on a per epoll context basis.

This makes epoll-based busy polling much more usable for user
applications than the current system-wide sysctl and hardcoded budget.

To allow for this, two ioctls have been added for epoll contexts for
getting and setting a new struct, struct epoll_params.

ioctl was chosen vs a new syscall after reviewing a suggestion by Willem
de Bruijn [1]. I am open to using a new syscall instead of an ioctl, but it
seemed that:
  - Busy poll affects all existing epoll_wait and epoll_pwait variants in
    the same way, so new verions of many syscalls might be needed. It
    seems much simpler for users to use the correct
    epoll_wait/epoll_pwait for their app and add a call to ioctl to enable
    or disable busy poll as needed. This also probably means less work to
    get an existing epoll app using busy poll.

  - previously added epoll_pwait2 helped to bring epoll closer to
    existing syscalls (like pselect and ppoll) and this busy poll change

Threaded it through to:

- glibc
- uclibc-ng
- musl (patch sent, waiting)

git://sourceware.org / glibc.git / commit

summary | shortlog | log | commit | commitdiff | tree
(parent: 400bdb5) | patch

commit

Linux: Add epoll ioctls

```
author     Joe Damato <jdamato@fastly.com>
           Tue, 28 May 2024 17:37:06 +0000 (17:37 +0000)
committer  Noah Goldstein <goldstein.w.n@gmail.com>
           Tue, 4 Jun 2024 17:09:15 +0000 (12:09 -0500)
commit     92c270d32caf3f8d5a02b8e46c7ec5d9d0315158
tree       cd19ddbbfe7031daa99539773a5d18daa934c1db        tree
parent     400bdb5c85af5a52b3f5653357c9fca87f036bd3        commit | diff
```

```
Linux: Add epoll ioctls

As of Linux kernel 6.9, some ioctls and a parameters structure have been
introduced which allow user programs to control whether a particular
epoll context will busy poll.

Update the headers to include these for the convenience of user apps.

The ioctls were added in Linux kernel 6.9 commit 18e2bf0edf4dd
("eventpoll: Add epoll ioctl for epoll_params") [1] to
include/uapi/linux/eventpoll.h.

[1]: https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/diff/?h=v6.9&id=18e2bf0edf4dd

Signed-off-by: Joe Damato <jdamato@fastly.com>
Reviewed-by: Adhemerval Zanella <adhemerval.zanella@linaro.org>
```

And, added a man page!

ioctl_eventpoll(2) System Calls Manual

NAME

ioctl_eventpoll, EPIOCSPARAMS, EPIOCGPARAMS - ioctl() operations for epoll file descriptors

LIBRARY

Standard C library (libc, -lc)

SYNOPSIS

```
#include <sys/epoll.h>  /* Definition of EPIOC* constants */
#include <sys/ioctl.h>

int ioctl(int fd, EPIOCSPARAMS, const struct epoll_params *argp);
int ioctl(int fd, EPIOCGPARAMS, struct epoll_params *argp);

#include <sys/epoll.h>

struct epoll_params {
  uint32_t busy_poll_usecs;   /* Number of usecs to busy poll */
  uint16_t busy_poll_budget;  /* Max packets per poll */
  uint8_t  prefer_busy_poll;  /* Boolean preference  */

 /* pad the struct to a multiple of 64bits */
  uint8_t  __pad;  /* Must be zero */
};
```

Around this same time, an academic paper was published:

Kernel vs. User-Level Networking: Don't Throw Out the Stack with the Interrupts

PETER CAI, University of Waterloo, Canada
MARTIN KARSTEN, University of Waterloo, Canada

This paper reviews the performance characteristics of network stack processing for communication-heavy server applications. Recent literature often describes kernel-bypass and user-level networking as a silver bullet to attain substantial performance improvements, but without providing a comprehensive understanding of how exactly these improvements come about. We identify and quantify the direct and indirect costs of asynchronous hardware interrupt requests (IRQ) as a major source of overhead. While IRQs and their handling have a substantial impact on the effectiveness of the processor pipeline and thereby the overall processing efficiency, their overhead is difficult to measure directly when serving demanding workloads. This paper

5.2.3 Kernel Polling. Kernel polling does not suffer from the issues that IRQ packing and IRQ suppression face, because the decision whether to poll or enable interrupts is made automatically based on the application's workload. Its performance is strong in both maximum throughput and tail latency, as evident by Table 5 and Figure 5. While it is difficult to compare throughput numbers for specific tail latencies with this methodology, it is clear that kernel polling outperforms the vanilla configuration by at least 30%.

The paper was showing promising results, but would we see those in our production workloads?

Adding

SO_INCOMING_NAPI_ID

support was much trickier than I expected

At Fastly:

- We use an open source web server called h2o, which uses **SO_REUSEPORT**
- Our machines have 1 or 2 dual port NICS
- We run 1 process which listens on all NICs
- We listen on lots of ports (80, 443, …)
- We use Mellanox ConnectX-5 Ex NICs

Rough diagram

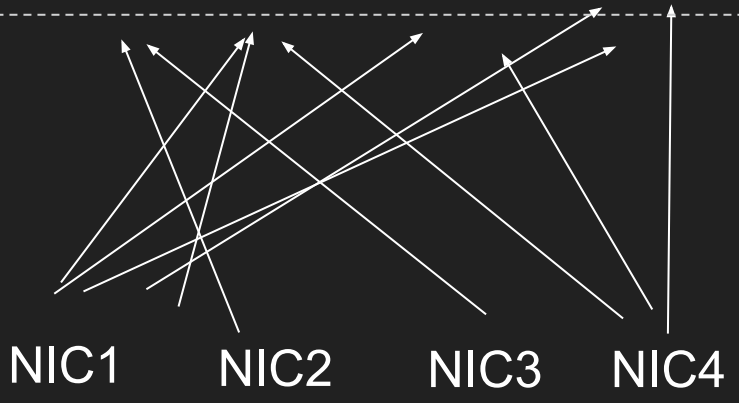
h2o threads

.....

Listening ports
All threads
Listen all ports
via
SO_REUSEPORT

80 80 80 80 — Reuseport group 1

443 443 443 443 — Reuseport group 2



Connections come in on any NIC and kernel balances amongst sockets in matching reusport group

I know what to do!

1. Custom RSS contexts to steer flows to CPUs where webserver worker threads run
2. Push IRQs out (we are going to poll)
3. Reuseport bpf filter inserted into the webserver code
4. Enable busy poll just for the web server via the ioctl !

The NIC settings via ethtool look like this

(sorry if this is small, download the slides later to read it)

```
NICS='eth1 eth2 eth3 eth4'

for DEV in ${NICS}; do
    # 16 queues per device
    sudo ethtool -L $DEV combined 16

    # for queues in the mask 0xff, turn off adaptive IRQ algorithms; we plan to busy poll so push IRQs out as far
    # as hardware allows
    sudo ethtool --per-queue $DEV queue_mask 0xff --coalesce adaptive-rx off adaptive-tx off
    sudo ethtool --per-queue $DEV queue_mask 0xff --coalesce rx-usecs 4095 rx-frames 65535 tx-usecs 4095 tx-frames 65535

    # create a custom RSS context which sends all flows to queue 0-7, this is context 1
    sudo ethtool -X $DEV weight 1 1 1 1 1 1 1 1 context new

    # set the default RSS context to send all flows to queues 8-15
    sudo ethtool -X $DEV weight 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1

    # attach a filter rule to send all flows with a dst-port of $PORT to the queue in RSS context 1 (e.g. queues 0-7)
    sudo ethtool -U $DEV flow-type tcp4 dst-port 80 context 1
    sudo ethtool -U $DEV flow-type tcp4 dst-port 443 context 1

    # attach a filter rule to send all flows with a dst-port of $PORT to the queue in RSS context 1 (e.g. queues 0-7)
    sudo ethtool -U $DEV flow-type tcp6 dst-port 80 context 1
    sudo ethtool -U $DEV flow-type tcp6 dst-port 443 context 1

    # pin IRQs to CPUs which are NUMA local to the devices (via lstopo --whole-io)
    if [ "${DEV}" = "vlan101" ]; then
        sudo ./set_irq_affinity 9,73,4,68,24,88,28,92,3,10,19,26,34,40,48,56 $DEV
    elif [ "${DEV}" = "vlan201" ]; then
        sudo ./set_irq_affinity 12,76,32,96,36,100,42,106,67,74,83,90,98,104,112,120 $DEV
    elif [ "${DEV}" = "vlan301" ]; then
        sudo ./set_irq_affinity 16,80,44,108,50,114,54,118,7,15,20,31,39,45,55,63 $DEV
    else
        sudo ./set_irq_affinity 22,86,58,122,62,126,64,82,71,79,84,95,103,109,119,127 $DEV
    fi

    # setup transmit steering based on receive-queues map
    # /sys/class/net/<dev>/queues/tx-<n>/xps_rxqs
    sudo ./set_xps_rxqs $DEV
done
```

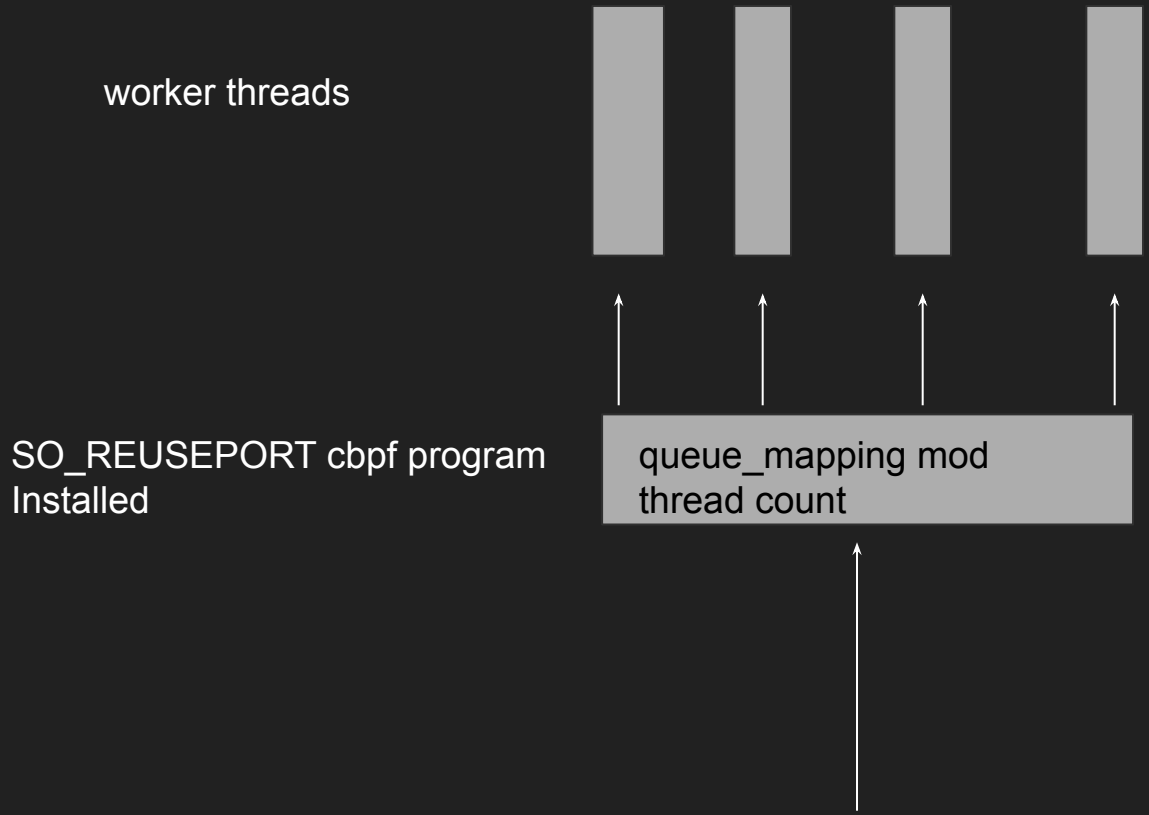
I know what to do!

1. ~~Custom RSS contexts to steer flows to CPUs where webserver worker threads run~~
2. ~~Push IRQs out (we are going to poll)~~
3. Reuseport bpf filter inserted into the webserver code
4. Enable busy poll just for the web server via the ioctl !

```
struct sock_filter code[] = {
    /* A = skb->queue_mapping */
    { BPF_LD | BPF_W | BPF_ABS, 0, 0, SKF_AD_OFF + SKF_AD_QUEUE },
    /* A = A % n */
    { BPF_ALU | BPF_MOD, 0, 0, n },
    /* return A */
    { BPF_RET | BPF_A, 0, 0, 0 },
};

struct sock_fprog p = {
    .len = ARRAY_SIZE(code),
    .filter = code,
};

if (setsockopt(fd, SOL_SOCKET, SO_ATTACH_REUSEPORT_CBPF, &p, sizeof(p))) {
```

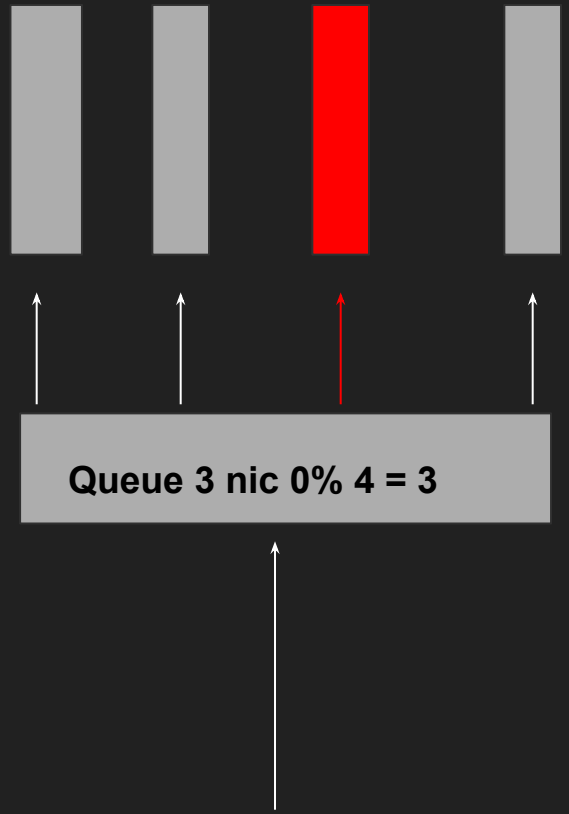
This actually doesn’t work when you have multiple NICs.

Buggy example



Buggy example part 1

worker
threads



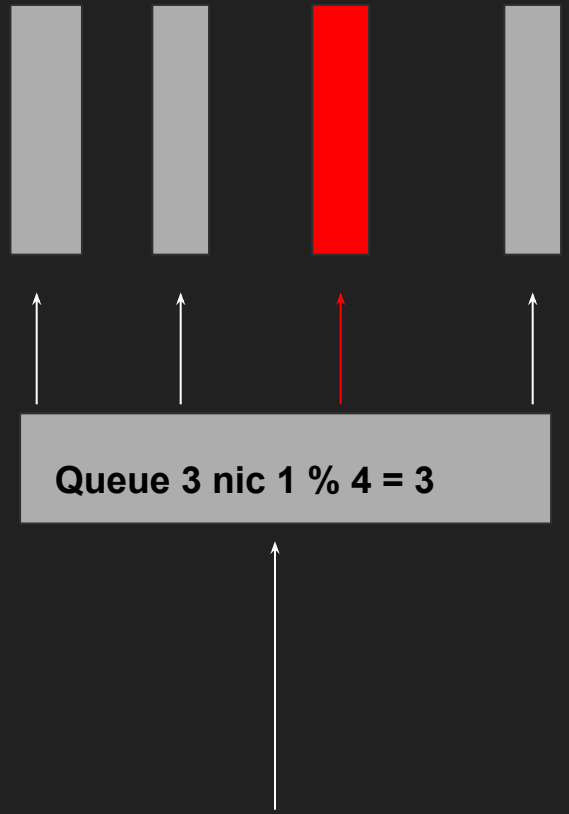
SO_REUSEPORT cbpf program

Connection with NAPI ID 1439
From RX queue 3 on NIC 0
arrives.

3 modulo 4 = 3
Thread 3 is selected

Buggy example part 2

worker
threads



SO_REUSEPORT cbpf program

Connection with NAPI ID 1443
From RX queue 3 on NIC 1
arrives.

3 modulo 4 = 3
Thread 3 is selected again

Now thread 3 has been given two connections with different NAPI IDs:

This breaks busy poll and is not allowed.

So… how to fix this?

A few ways come to mind, but this is what I did

Read a lot of kernel code

```
static int inet_reuseport_add_sock(struct sock *sk,
                                   struct inet_listen_hashbucket *ilb)
{
        struct inet_bind_bucket *tb = inet_csk(sk)->icsk_bind_hash;
        const struct hlist_nulls_node *node;
        struct sock *sk2;
        kuid_t uid = sock_i_uid(sk);

        sk_nulls_for_each_rcu(sk2, node, &ilb->nulls_head) {
                if (sk2 != sk &&
                    sk2->sk_family == sk->sk_family &&
                    ipv6_only_sock(sk2) == ipv6_only_sock(sk) &&
                    sk2->sk_bound_dev_if == sk->sk_bound_dev_if &&
                    inet_csk(sk2)->icsk_bind_hash == tb &&
                    sk2->sk_reuseport && uid_eq(uid, sock_i_uid(sk2)) &&
                    inet_rcv_saddr_equal(sk, sk2, false))
                        return reuseport_add_sock(sk, sk2,
                                                  inet_rcv_saddr_any(sk));
        }

        return reuseport_alloc(sk, inet_rcv_saddr_any(sk));
}
```

What if… listen sockets were grouped per NIC with SO_BINDTODEVICE?

Rough diagram

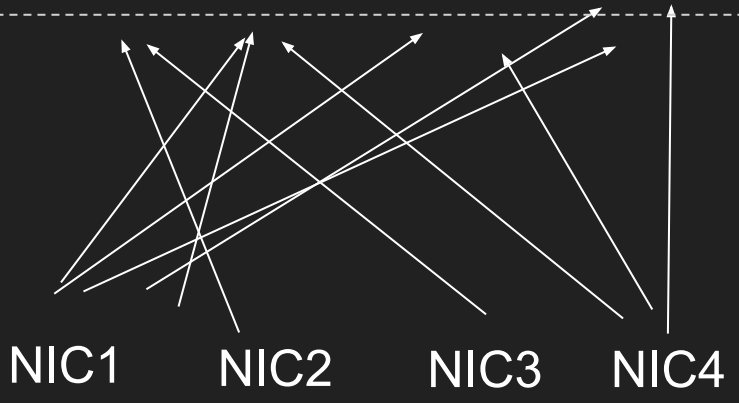
h2o threads

.....

Listening ports
All threads
Listen all ports
via
SO_REUSEPORT

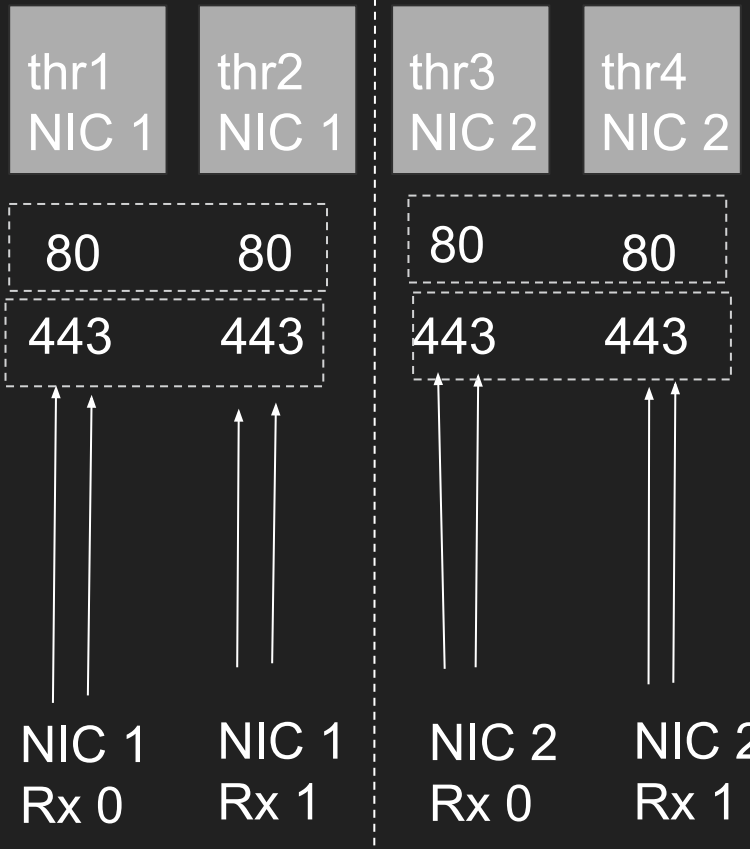
80 80 80 80 — Reuseport group 1

443 443 443 443 — Reuseport group 2



Connections come in on any NIC and kernel balances amongst sockets in matching reusport group

Busy poll h2o



The code to get there is reasonable

Rough pseudo code of how I approached this:

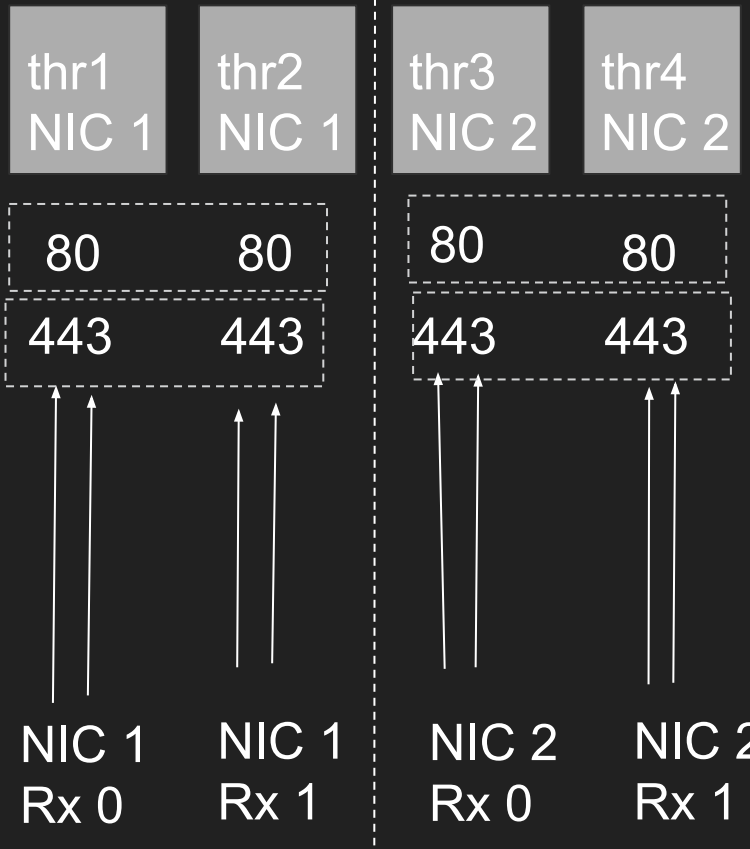
```
char *ifaces[2] = ["eth0", "eth1"];
struct iface_listen_sockets listen_sockets[2];
char *iface;
int thread_count = 4;
int reuse = 1;
int *fd;

/* create 1 listen socket per worker thread per iface */
for (int iff = 0; iff < 2; iff++) {
        iface = ifaces[iff];
        listen_sockets = listen_sockets[iff];

        for (int i = 0; i < thread_count; i++) {
                /* create the socket */
                listen_sockets->fd[i] = socket(domain, type, protocol);

                /* bind it to the device */
                setsockopt(listen_sockets->fd[i], SOL_SOCKET, SO_BINDTODEVICE, iface, strlen(iface));

                /* set reuseport */
                setsockopt(listen_sockets->fd[i], SOL_SOCKET, SO_REUSEPORT, &reuse, sizeof(reuse));

                /* apply the BPF filter to the first socket in the group */
                if (i == 0) {
                        apply_bpf_filter(fd, thread_count);
                }
        }
}
```

```
/* now that the reuseport groups are created, call listen on all
 * the sockets
 */
for (int iff = 0; iff < 2; iff++) {
        listen_sockets = listen_sockets[iff];

        for (int i = 0; i < thread_count; i++) {
                listen(listen_sockets->fd[i], 1024);
        }
}
```

Turned out this worked!

So now, incoming connections are going to worker threads while maintaining the constraint that each thread maps to one NAPI ID.

Busy poll h2o



I think it’s important to mention what the accept path looks like.

This is what I did, but there many other ways, too!

App config read at startup maps NICs to NUMA local CPUs

```
busy-poll-cpu-map:
  eth1:
    - 9
    - 10
    - 11
    - ....
  eth2:
    - 18
    - ....
```

So, accept path looks like this (psuedo code):

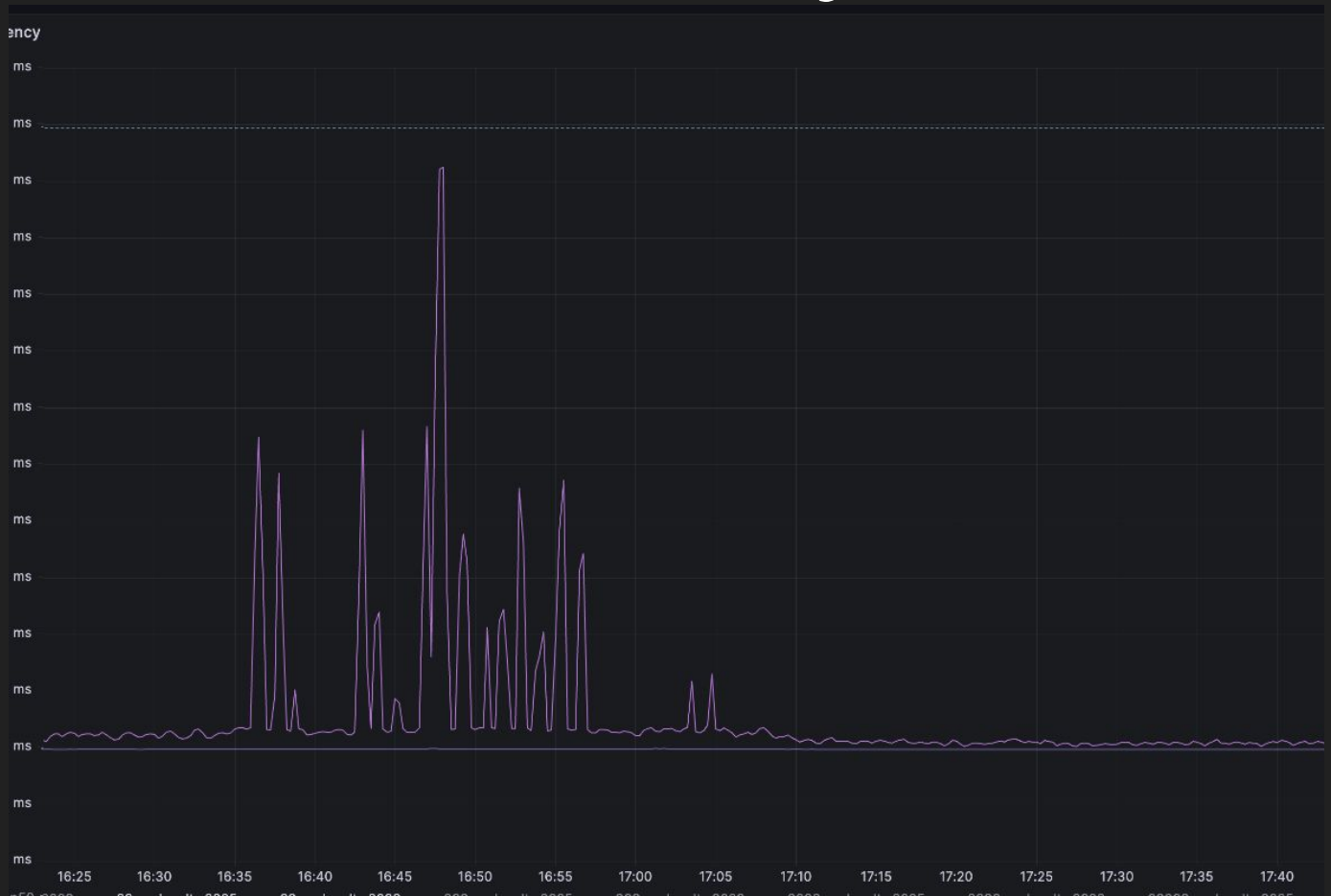
```c
static __thread int cpu_claimed = 0;
static __thread int napi_id_claimed = 0;

static void handle_busy_poll_accept(int sockfd)
{
        unsigned napi_id = get_napi_id(sockfd);

        /* has this thread claimed a CPU? */
        if (cpu_claimed) {
                /* it has... does the claimed NAPI ID match the new
                 * connection?
                 */
                if (napi_id_claimed == napi_id)
                        return;

                /* here we have a mismatch. this is what happens if you
                 * have multiple NICs and you dont setup the reuseport
                 * groups in the kernel correctly
                 */
                if (napi_id_claimed != napi_id)
                        fire_an_alert();
        } else {
                struct ifreq ifr = {};

                /* use a hack to get the NIC name from NAPI ID */
                get_nic_name_by_napi(sockfd, napi_id, &ifr);

                /* pick an un-used CPU from the list provided in config */
                int cpu_id = assign_cpu_from_map(ifr.ifr_name);

                /* thread claims that CPU and NAPI ID */
                cpu_claimed = cpu_id;
                napi_id_claimed = napi_id;

                /* pin the thread to that CPU*/
                pin_thread_to_cpu(cpu_id);
        }
```

Internal hack that we use:

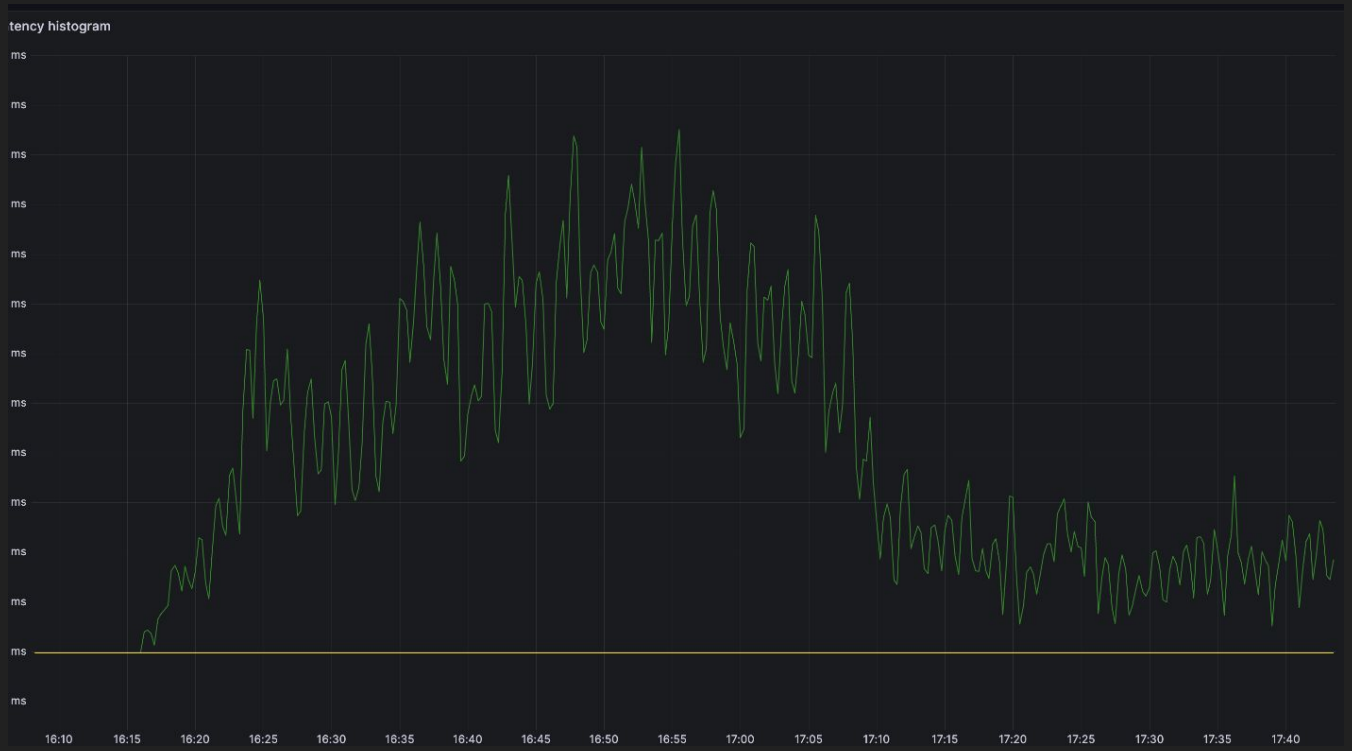
```
/* this is an internal kernel hack; works like SIOCGIFNAME but given a NAPI
 * ID (instead of an ifindex) fills in the interface name.
 *
 * instead, drivers should implement netdev-genl to map NAPI IDs to
 * ifindexes
 *
 * then user apps can use that real interface instead of this hack
 */
static int get_nic_name_by_napi(int fd, unsigned int napi_id, struct ifreq *ifr)
{
    ifr->ifr_ifru.ifru_ivalue = napi_id;

    return ioctl(fd, SIOCGIFNAME_BY_NAPI_ID, ifr);
}
```

OK …. So we did all that, what does it look like in prod?

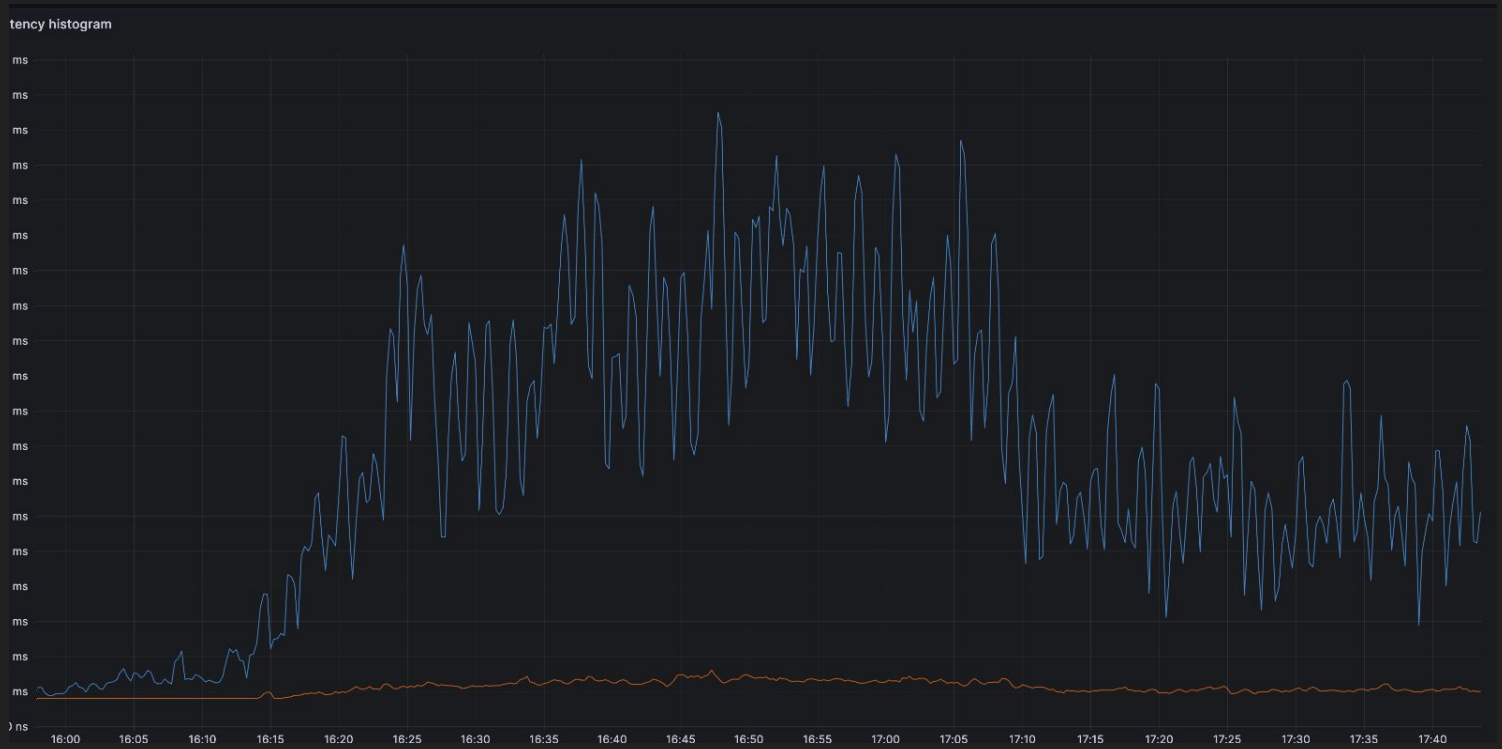
P99 latency better



P999 latency better



P9999 latency better

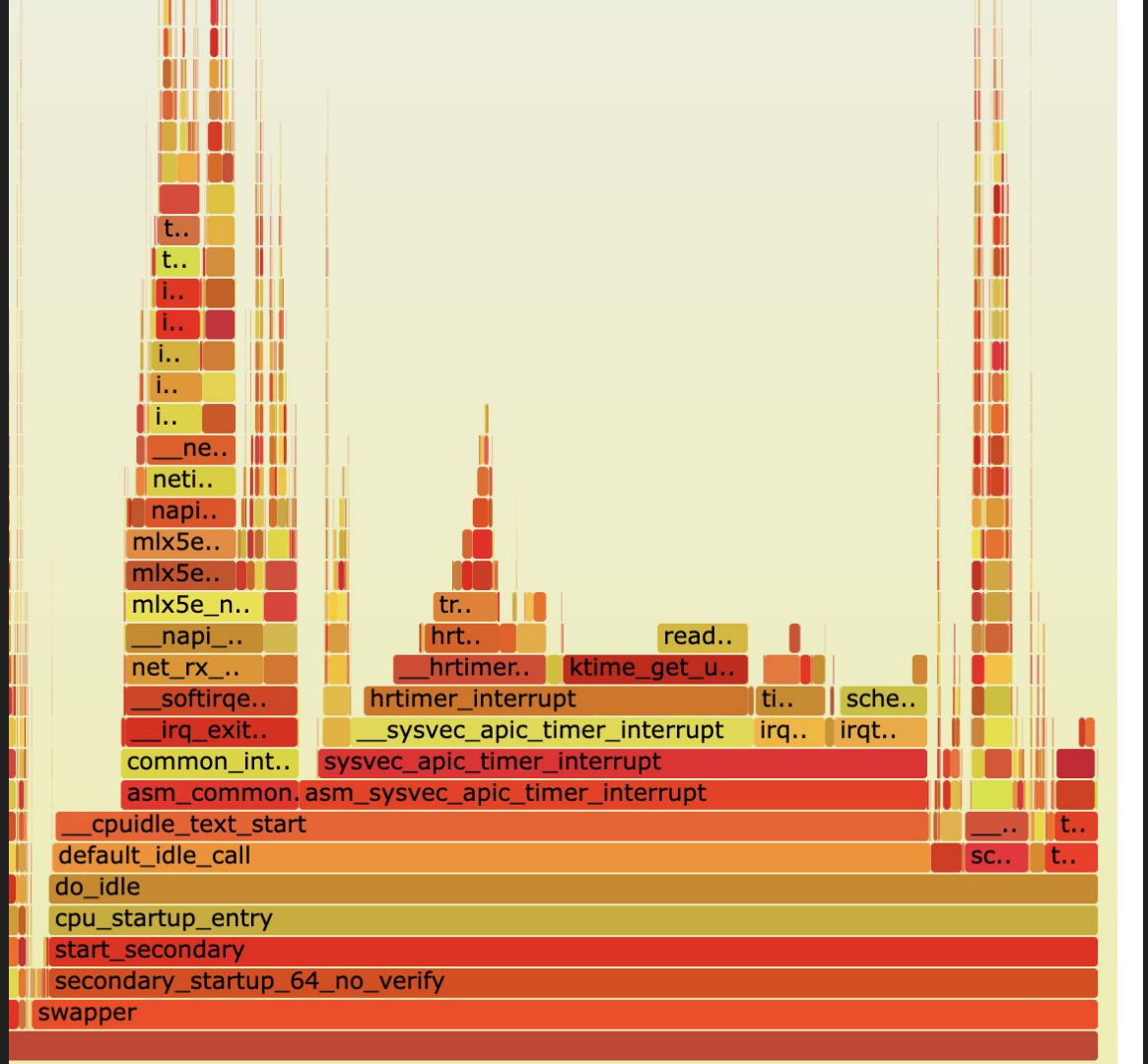


Cool, so latency looks much better for the higher percentiles.

BUT

1. we use epoll_pwait2
2. we set the busy poll usecs to match the pwait2 time
3. Occasionally, pwait2 runs idle

perf measurement suggests IRQs from the device still arrive which kick off NAPI and induce latency


t..
t..
i..
i..
i..
i..
i..
__ne..
neti..
napi..
mlx5e..
mlx5e..
mlx5e_n..
__napi_..
net_rx_..
__softirqe..
__irq_exit..
common_int..
asm_common..
tr..
hrt..
__hrtimer..
hrtimer_interrupt
__sysvec_apic_timer_interrupt
sysvec_apic_timer_interrupt
asm_sysvec_apic_timer_interrupt
read..
ktime_get_u..
ti..
irq..
sche..
irqt..
__cpuidle_text_start
default_idle_call
do_idle
cpu_startup_entry
start_secondary
secondary_startup_64_no_verify
swapper
__..
sc..
t..
t..

Can we maybe set maxevents lower and defer IRQs with:

napi_defer_hard_irqs
gro_flush_timeout

We can’t.

These settings are system-wide. Other latency sensitive apps that are not busy-poll compatible will suffer.

Turns out choosing these two values is really, really hard anyway.

That’s where the future comes in:

Some things you may want to work on
Some things I and Fastly am/are working on

You (driver maintainers) can:

- Add netdev-genl support to your driver
- Already supported by:
  - mlx4
  - mlx5
  - ice
  - bnxt

I am currently working on:

A collaboration with Martin Karsten of U Waterloo:

- A kernel patch which:
  - disables interrupts during busy poll **but**
  - re-enables IRQs if busy poll has no data
  - Avoids needing to pick “the right numbers” for napi_defer_hard_irqs and gro_flush_timeout which might change if the system is under load or idle
  - gro_flush_timeout serves as a “backstop” if userland is slow

- Initial data measured against memcached looks *very promising*.

- Enabled via a busy poll ioctl per context

Hopefully submitting an RFC with data soon.

Fastly is working on

Upstreaming our busy poll implementation to open source h2o.

Not ready yet, but we are working on it and planning to open source it.

I am *hoping* to work on:

napi_defer_hard_irqs
gro_flush_timeout

per NAPI

(via netdev-genl hopefully?)