第二届CCF·夜莺开发者创新论坛

中国北京　2024.7.26

主办方：中国计算机学会 | 承办方：CCF开源发展委员会、夜莺项目开源社区

自我介绍

- 2016 年北京大学毕业 电子通信工程

- 2017 年入职滴滴 目前基础平台系统软件负责人

- 主要聚焦 Linux 内核技术，容器技术，云网络技术。持续向Linux 内核，Open vSwitch 社区贡献200+Pr。

eBPF 在内核可观测故障定位中的实践

张同浩 滴滴 2024.07.26

中国北京　2024.7.26

主办方：中国计算机学会 | 承办方：CCF开源发展委员会、夜莺项目开源社区

大纲

1 系统故障定位的现状和挑战

2 我们的方案和演进

3 故障定位的应用实践

系统故障定位的现状和挑战

- 观测手段
- 故障现场
- 偶发毛刺

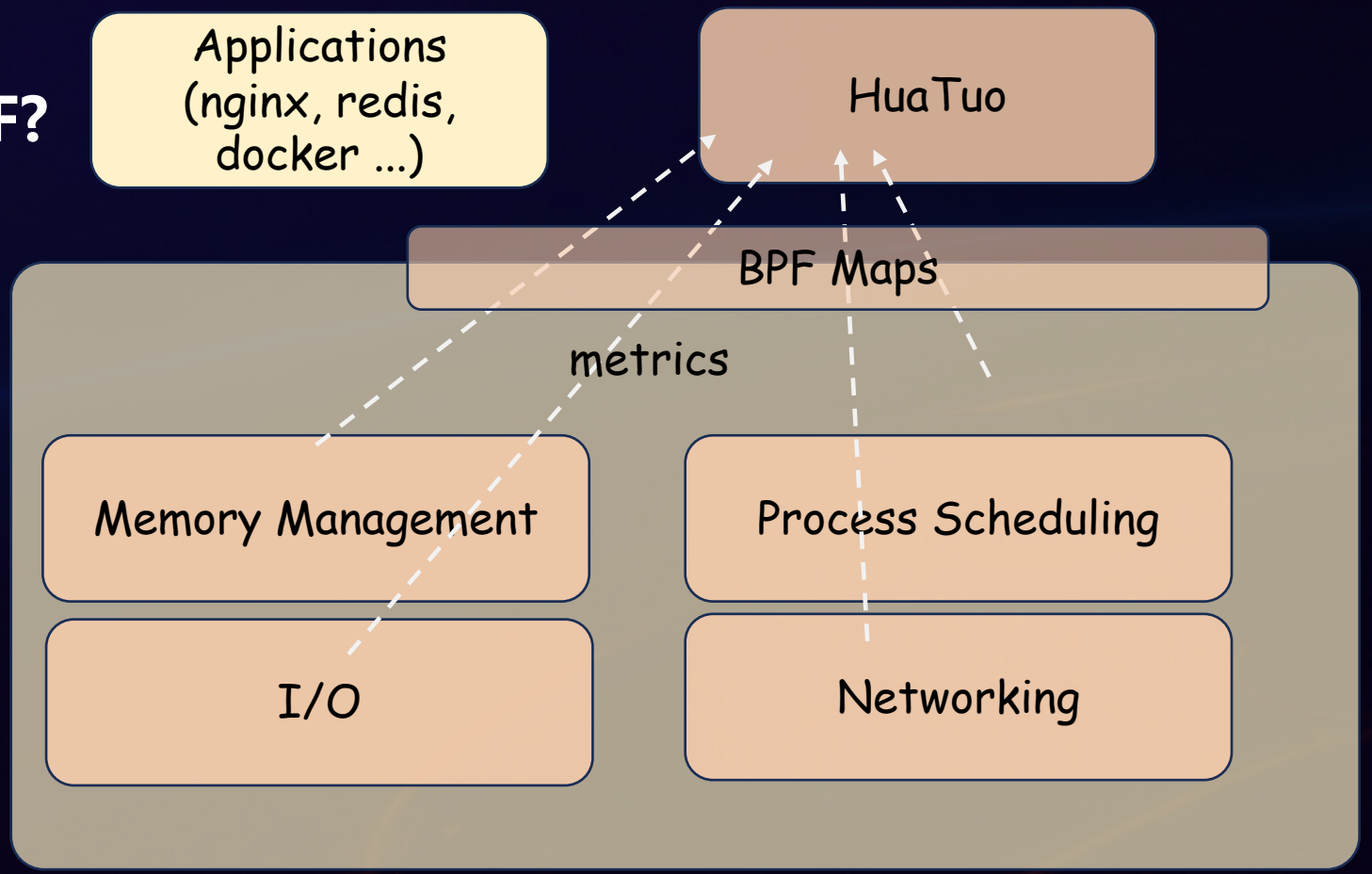
系统故障定位的现状和挑战

- 业界方案
  - Alibaba SysAK
  - Pixie: Kubernetes Monitoring, Application Debug Platform
  - Bcc/kernelshark/trace-cmd/perf-tools ...

我们的方案和演进

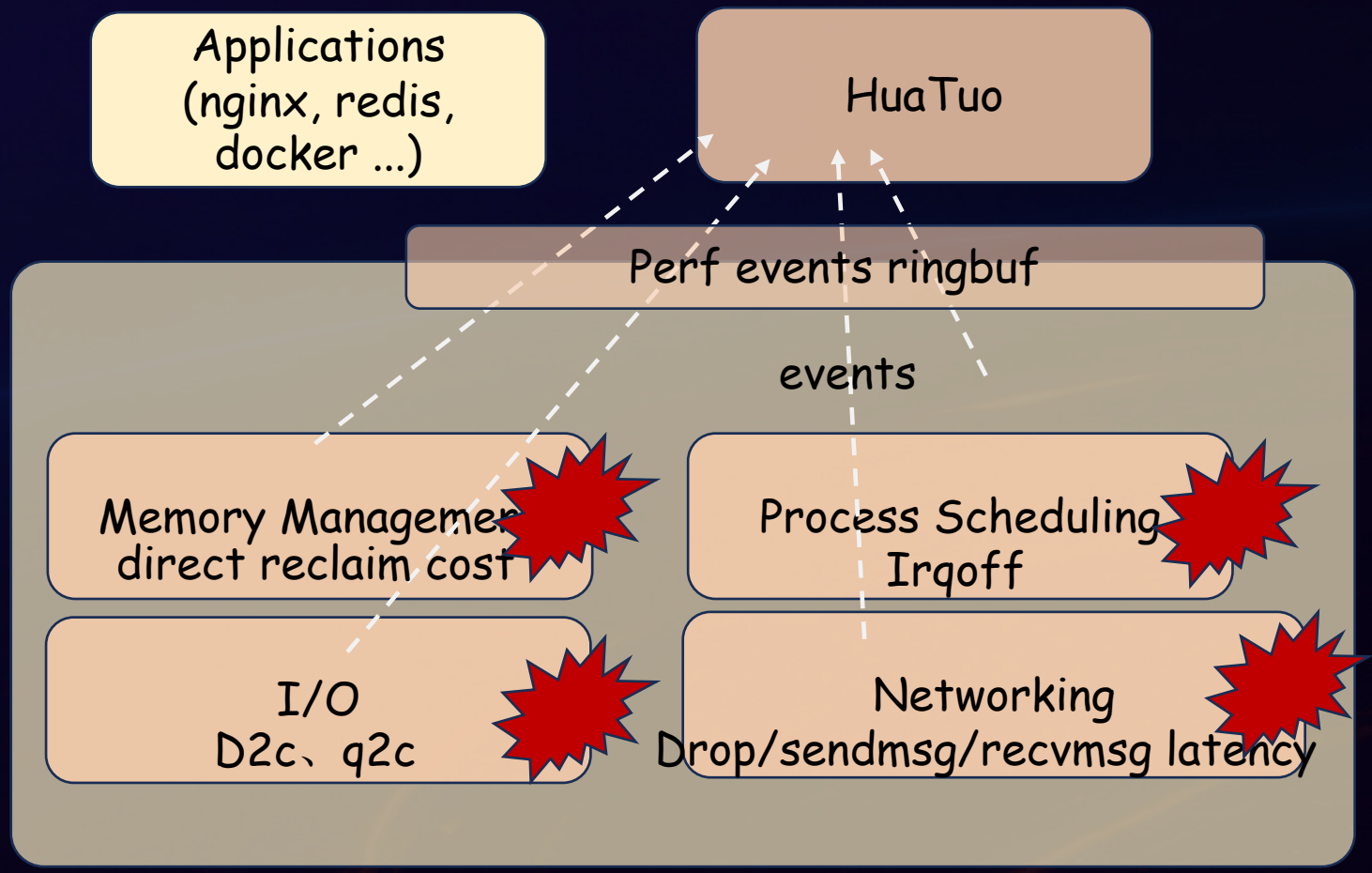
- 关键指标
  - 修改内核？内核模块？eBPF?
  - 埋点
  - BPF map
  - 限速 降级

subsystem	metrics
memory	Direct/kswapd latency
Cpu sched	Runqlat
IO	D2c, Q2C, xfs log
networking	xmit , qdisc latency , drop



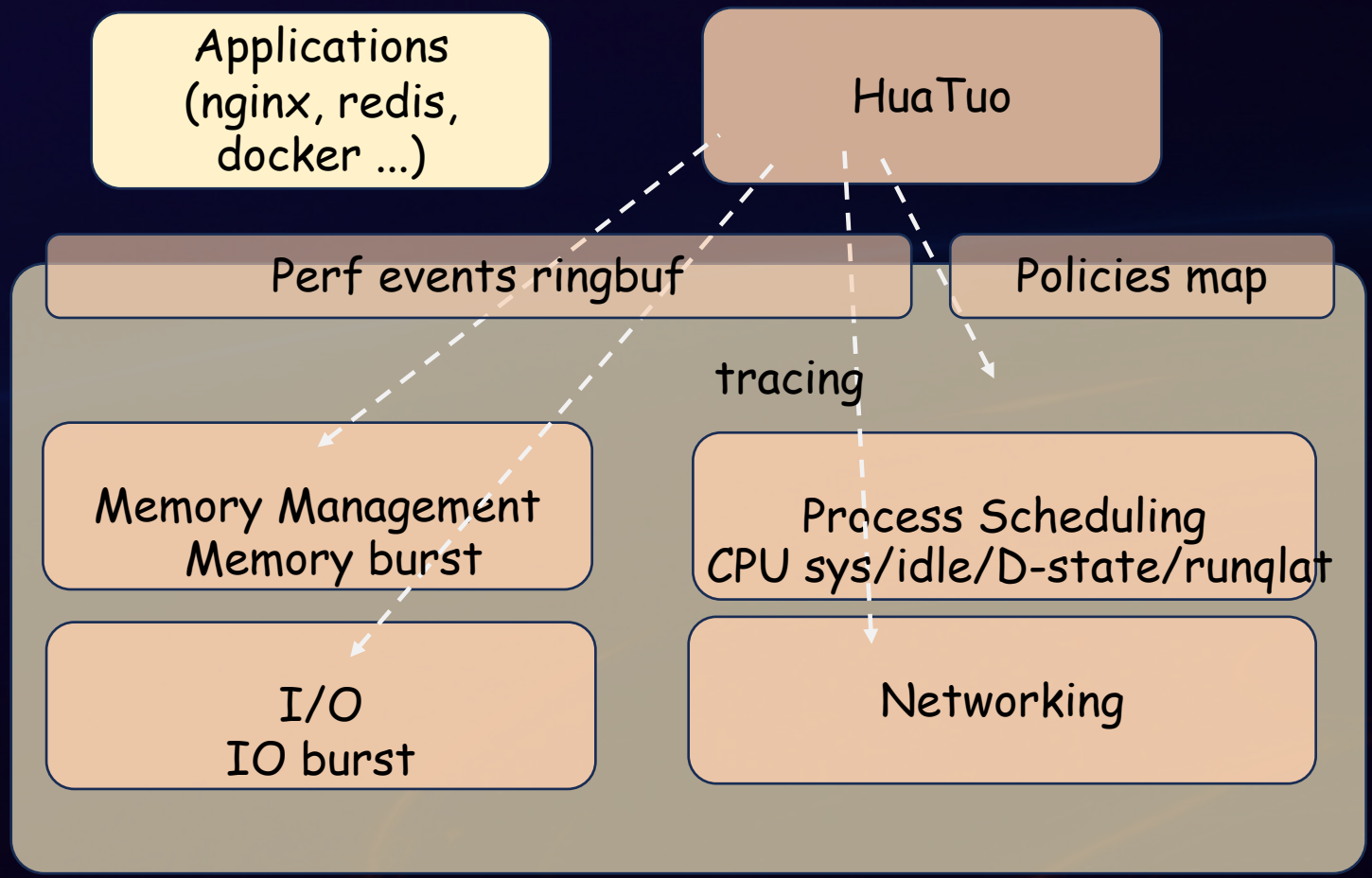
我们的方案和演进

- 关键事件
  - 事件驱动 阈值触发
  - 关键执行上下文
  - 常态运行 5%



我们的方案和演进

- AutoTracing
  - 全栈 AutoTracing
  - 指标驱动 + 关键工具



我们的方案和演进

- Profiling
  - Perf cpu_clock + eBPF
  - Host/Containers/Process
  - Golang/C++/Java

我们的方案和演进

- 解决观测维度

观测维度不同，损耗不同，有效上下文不同，适用场景不同
-- 相互配合
-- 因地适宜

我们的方案和演进

Metrics

细粒度系统运行状态

Events

关键执行上下文

AutoTracing

更加丰富执行上下文
应对偶发毛刺

Profiling

持续应用性能分析

我们的方案和演进

- 开源标准化

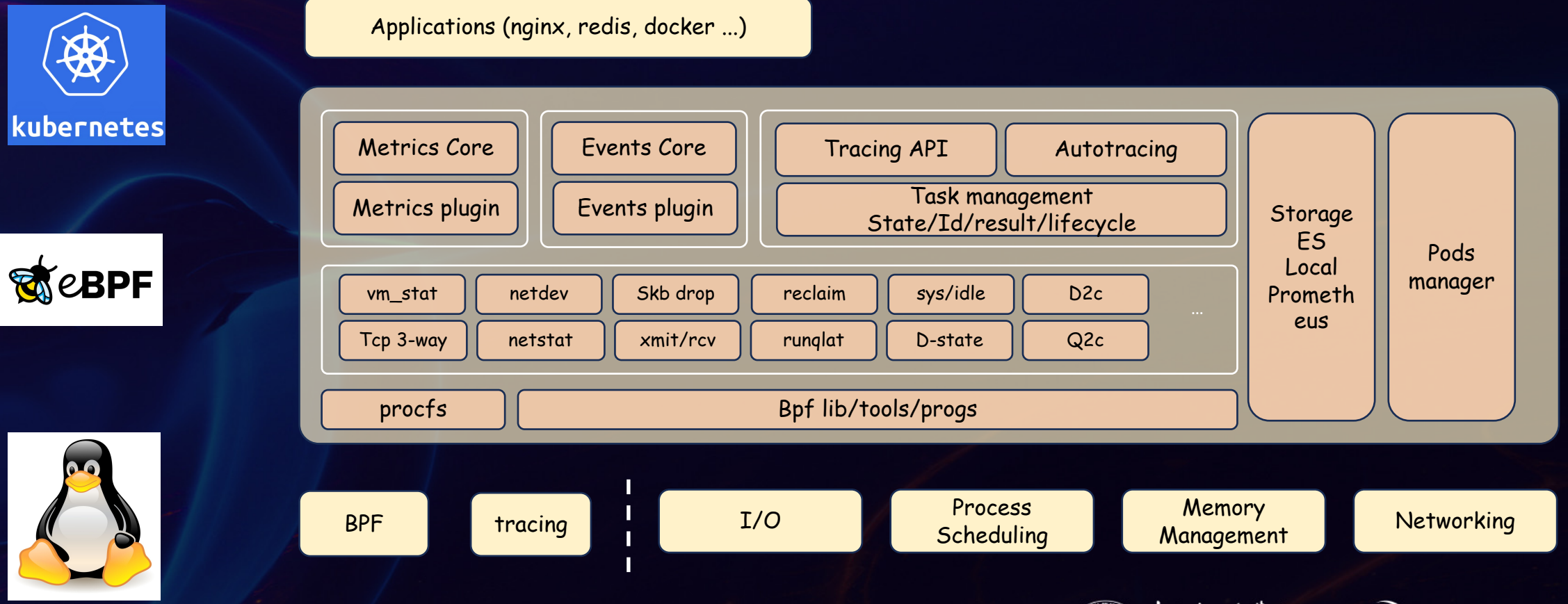
- 逻辑机房
- 逻辑集群
- 指标表达
- 组件依赖：odin, kube-agent, irmas 等

剥离公司特定服务依赖

- **Prometheus Metrics**
  独立数据采集器 对开源观测体系的补充
- **Elastic Search**
  日志，Tracing，火焰图数据
- **Grafana UI**
  前端监控展示
- **K8S Pods/Client-go**
  容器信息

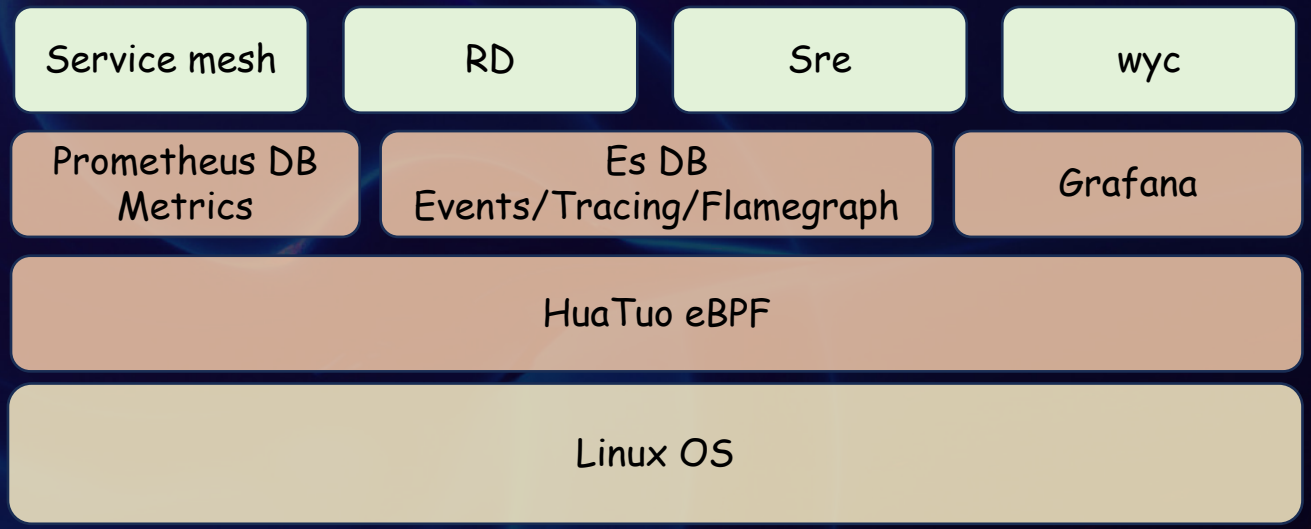
我们的方案和演进

- 整体架构



我们的方案和演进

- 我们在哪？用户是谁? 走向何处？



- 基础平台的基座
- 集团产研：基础研发，业务研发，Sre，第三方
- 分布式Tracing、业务和系统观测数据打通串联

故障定位的应用

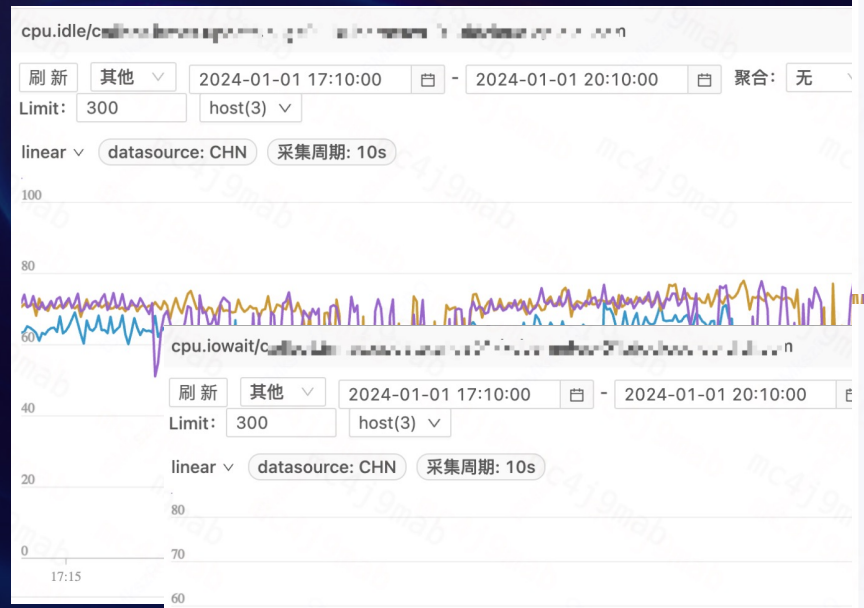
- Irqoff 异常事件

```
stack backtrace:
        run_timer_softirq+0x1 [kernel]
        __softirqentry_text_start+0xe3 [kernel]
        irq_exit+0x100 [kernel]
        smp_apic_timer_interrupt+0x74 [kernel]
        apic_timer_interrupt+0xf [kernel]
        read_hpet+0x38 [kernel]
        ktime_get_ts64+0x40 [kernel]
        posix_ktime_get_ts+0xd [kernel]
        __x64_sys_clock_gettime+0x59 [kernel]
        do_syscall_64+0x5b [kernel]
        entry_SYSCALL_64_after_hwframe+0x65 [kernel]
```

```
791 static u64 read_hpet(struct clocksource *cs)
792 {
793         unsigned long flags;
794         union hpet_lock old, new;
795
796         BUILD_BUG_ON(sizeof(union hpet_lock) != 8);
797
798         /*
799          * Read HPET directly if in NMI.
800          */
801         if (in_nmi())
802                 return (u64)hpet_readl(HPET_COUNTER);
803
804         /*
805          * Read the current state of the lock and HPET value ato
806          */
807         old.lockval = READ_ONCE(hpet.lockval);
808
809         if (arch_spin_is_locked(&old.lock))
810                 goto contended;
811
812         local_irq_save(flags);
813         if (arch_spin_trylock(&hpet.lock)) {
814                 new.value = hpet_readl(HPET_COUNTER);
815                 /*
816                  * Use WRITE_ONCE() to prevent store tearing.
817                  */
818                 WRITE_ONCE(hpet.value, new.value);
819                 arch_spin_unlock(&hpet.lock);
820                 local_irq_restore(flags);
821                 return (u64)new.value;
822         }
823         local_irq_restore(flags);
```

故障定位的应用

- 系统 idle 掉底




```
        "fs_read": 44544,
        "disk_read": 17269760,
        "comm": "bin/kube-agent --config conf/kube-agent.yaml ",
        "pid": 3729,
        "file_stat": [
          "[253:17], fs_read=39424b/s, fs_write=0b/s, disk_read=17268736b/s, disk_write=0b/s, q2c=1008us, d2c=886us, inode=1066186, didicloud/kube-agent/bin/kube-agent",
          "[253:17], fs_read=5120b/s, fs_write=765b/s, disk_read=1024b/s, disk_write=1536b/s, q2c=309us, d2c=305us, inode=3222236004, didicloud/kube-agent/logs/kube-agent.log"
        ],
        "fs_write": 765,
        "ct_hostname": "",
        "disk_write": 1536
      },
      {
        "fs_read": 760320,
        "disk_read": 12179456,
        "comm": "/usr/local/gundam/gundam_client/client -type=agent ",
        "pid": 9490,
        "file_stat": [
          "[253:1], fs_read=34304b/s, fs_write=0b/s, disk_read=1536b/s, disk_write=0b/s, q2c=8632us, d2c=8627us, inode=142695, etc/passwd",
          "[253:1], fs_read=33280b/s, fs_write=0b/s, disk_read=12176384b/s, disk_write=0b/s, q2c=110305us, d2c=107200us, inode=1444017, local/gundam/gundam_client/client",
          "[253:1], fs_read=692736b/s, fs_write=0b/s, disk_read=1536b/s, disk_write=0b/s, q2c=331us, d2c=327us, inode=138864, etc/group"
        ],
        "fs_write": 0
```

故障定位的应用

- Skb drop

comm	poi-search
daddr	43.139.252.204
dest_hostname	<nil>
dport	443
max_ack_backlog	0
pid	3473236
pkt_len	74
queue_mapping	30
saddr	10.212.232.161
seq	1708012458
sport	47086
src_hostname	poi-search-phnc-sf-778a8-51.docker.qgz03.diditaxi.com.
stack	poi-search 3473236 kfree_skb/ffffffff9070d7b0 kfree_skb/ffffffff9070d7b0 kfree_skb_list/ffffffff9070d850 __dev_queue_xmit/ffffffff90724530 ipvlan_queue_xmit/ffffffffc0b55fa0 ipvlan_start_xmit/ffffffffc0b567d0 dev_hard_start_xmit/ffffffff907242a0 sch_direct_xmit/ffffffff90775ad0 __qdisc_run/ffffffff90775df0 __dev_queue_xmit/ffffffff90724530 ip_finish_output2/ffffffff9079e120 ip_output/ffffffff9079fc90 __tcp_transmit_skb/ffffffff907b99c0 tcp_connect/ffffffff907ba4a0 tcp_v4_connect/ffffffff907c1030 __inet_stream_connect/ffffffff907daa90 inet_stream_connect/ffffffff907dae00 __sys_connect/ffffffff90702d30 __x64_sys_connect/ffffffff90702e30 do_syscall_64/ffffffff90004140 entry_SYSCALL_64_after_hwframe/ffffffff90a00048

comm	swapper/37
daddr	10.188.253.191
dest_hostname	<nil>
dport	8055
max_ack_backlog	0
pid	0
pkt_len	60
queue_mapping	0
saddr	10.188.253.195
seq	3954012851
sport	26474
src_hostname	gift-proxy-sf-e41b5-37.docker.nmg01.diditaxi.com.
stack	swapper/37 0 kfree_skb/ffffffffa4b047b0 kfree_skb/ffffffffa4b047b0 neigh_invalidate/ffffffffa4b2a8b0 neigh_timer_handler/ffffffffa4b2c870 call_timer_fn/ffffffffa4536a70 run_timer_softirq/ffffffffa4537410 __softirqentry_text_start/ffffffffa5000000 irq_exit/ffffffffa44b7740 smp_apic_timer_interrupt/ffffffffa4e02720 apic_timer_interrupt/ffffffffa4e01ce0 poll_idle/ffffffffa4c9dad0 cpuidle_enter_state/ffffffffa4aaf930 cpuidle_enter/ffffffffa4aafd70 do_idle/ffffffffa44e6c20 cpu_startup_entry/ffffffffa44e7010 start_secondary/ffffffffa444ea10 secondary_startup_64/ffffffffa4400030
state	SYN_SENT
time	2024-07-16 19:28:31.951 +0800
tracer_name	dropwatch

感谢聆听

Thank you for listening