第三届 eBPF开发者大会

www.ebpftravel.com

基于ebpf的可观测工具实践分享

任金林，华为GTS&三丫坡服务战队

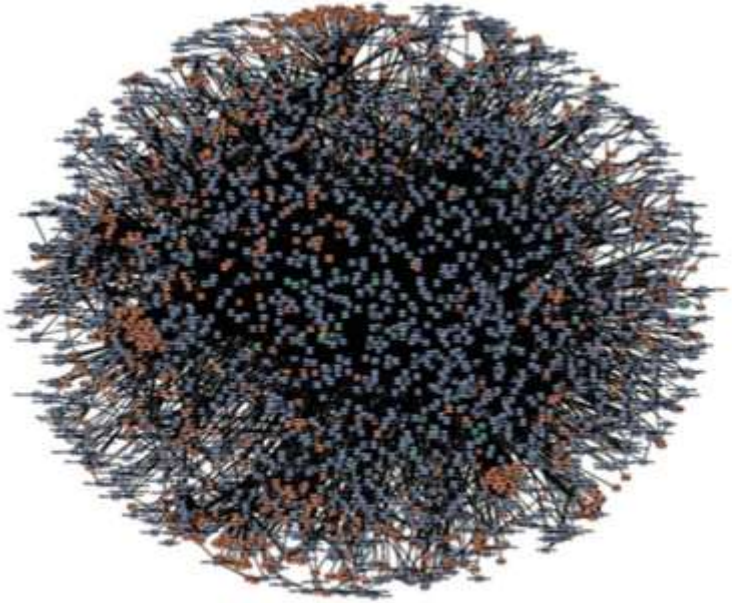
中国·西安

云原生系统结构复杂，海量微服务，故障难以快速定界定位，MTTR挑战巨大

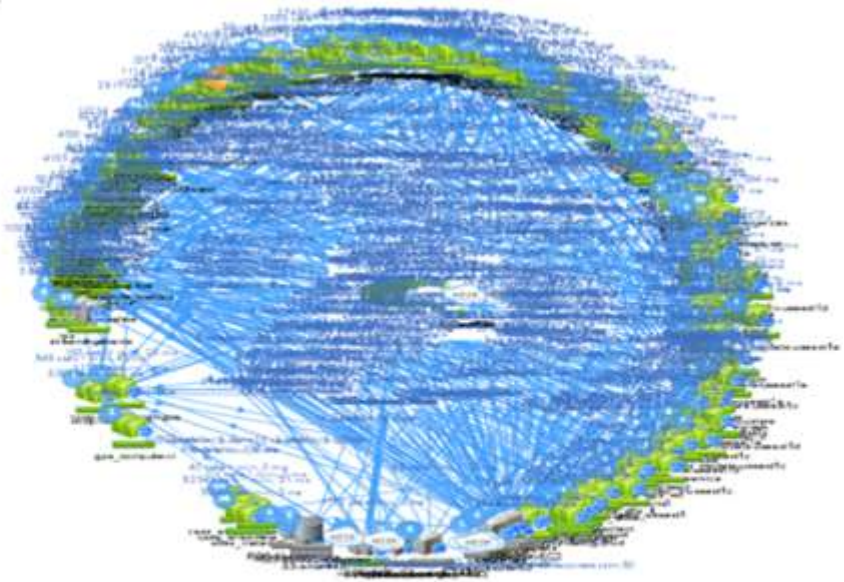
- 随着云服务的兴起，云原生软件系统发展迅速，规模呈现指数级增加，动辄上千个微服务，例如：Netflix **700+种微服务**， amazon有**3000+种微服务**， 华为GTS产品的**微服务1000+种**……；
- 微服务之间呈现复杂的调用关系，调用链跨越**几十种微服务**；

硬件设备上万

微服务成百上千，应用实例上百万



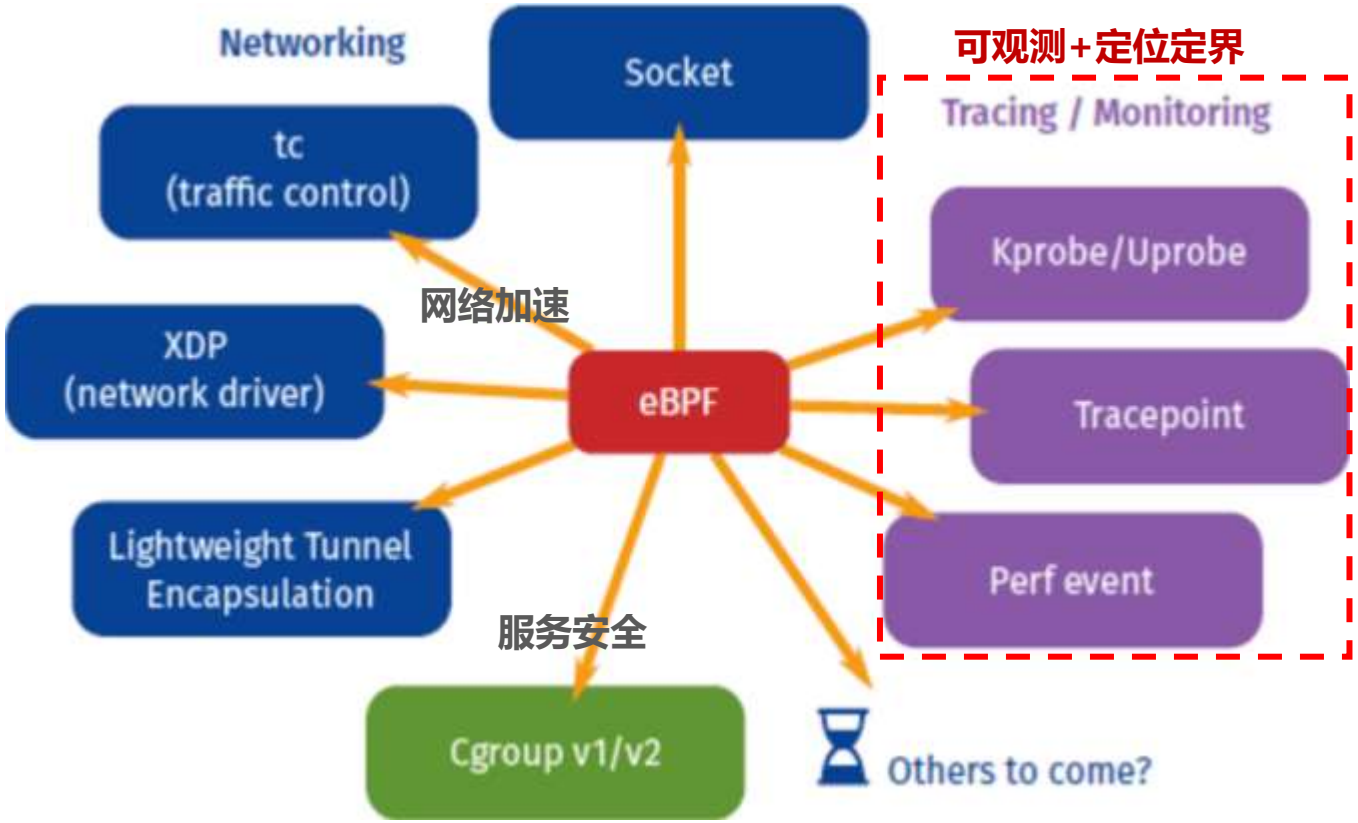
amazon.com



NETFLIX

eBPF - 解题方法实践介绍

eBPF技术成熟，具备从**内核层**到用户层所有指标数据的全面可观测，可以实现**无侵入式**的网络加速、服务安全、**全链路追踪、性能分析、函数级调用回溯**等能力



eBPF-探针实现

探针观测范围：300+Metrics（内核、容器、运行时、基础中间件、应用等）

Applications C / Java / python / php / ruby / node RPC（ TxRx统计 / 吞吐量 / 时延 / 跟踪 / 错误率）

进程（I/O操作分段统计/系统调用失败/ BIO操作失败/ I/O 挂死 / 状态异常）

火焰图 （cpu / MEM / 调度 / I/O等）

Runtimes JVM (OOM / Heap /缓存 / GC / lock / Mem)

DB （慢sql / 状态异常 / 死锁 ）

System Libraries 内存泄漏 / ssl监控 / 系统时延

Container

- CPU
- Mem
- Network
- I/O
- Process

System Call interface 系统调用统计 / 系统调用异常 / 系统调用时长 / 网络请求 / kill事件 / 内存申请 / 文件操作 / CGroup

VFS 文件缓存/ 挂载监控 / 文件操作统计

Sockets 连接状态 / 队列 / 错误统计

File Systems (Ext4/ nfs /Btrfs) 状态 / 时延 / 错误

TCP/UDP 丢包 / 重传 / 乱序 / 时延 / 缓存

Scheduler 延迟 / 状态迁移 / 软中断 运行队列 / 死锁

Volume Manager 状态监控 / 错误统计

IP 分片 / 路由转发 / 协议跟踪

Block Device 读 / 写 / 时延 / 队列

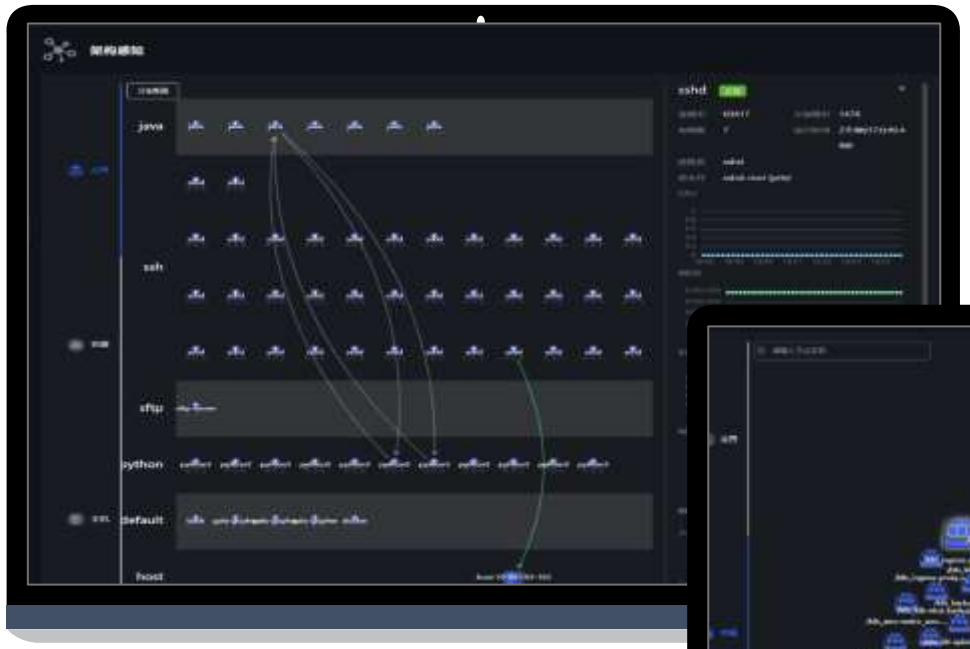
Net Device 丢包 / 错包 / 拥塞 / 队列 / 缓存

Virtual Memory oom-kill / 内存泄漏 共享内存 / 交换内存 脏页 / slabinfo

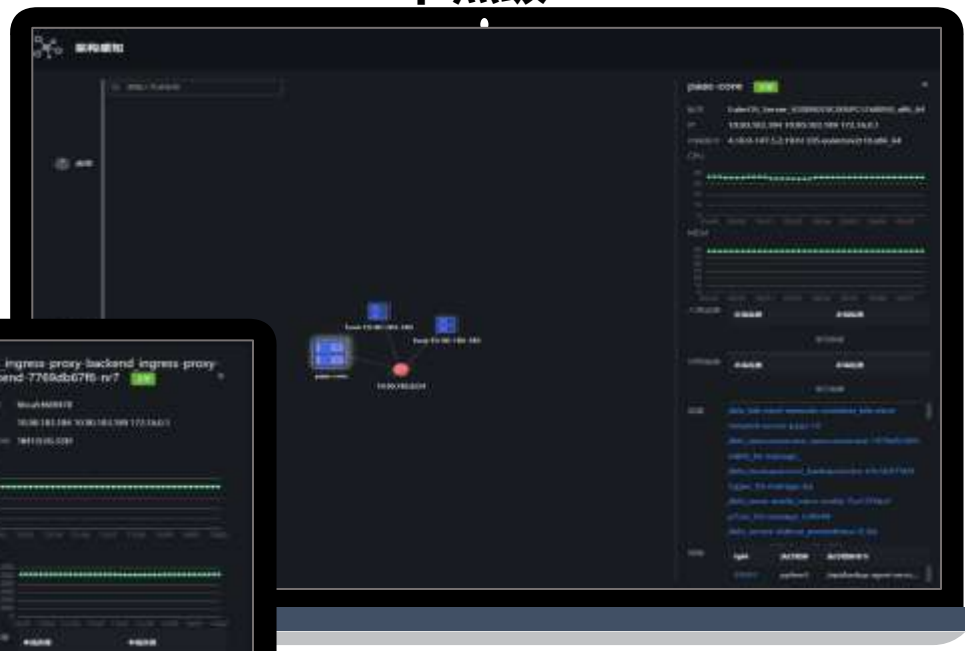
Device Drivers 中断 / 缓存 / 错误 / 连接

架构感知

进程级



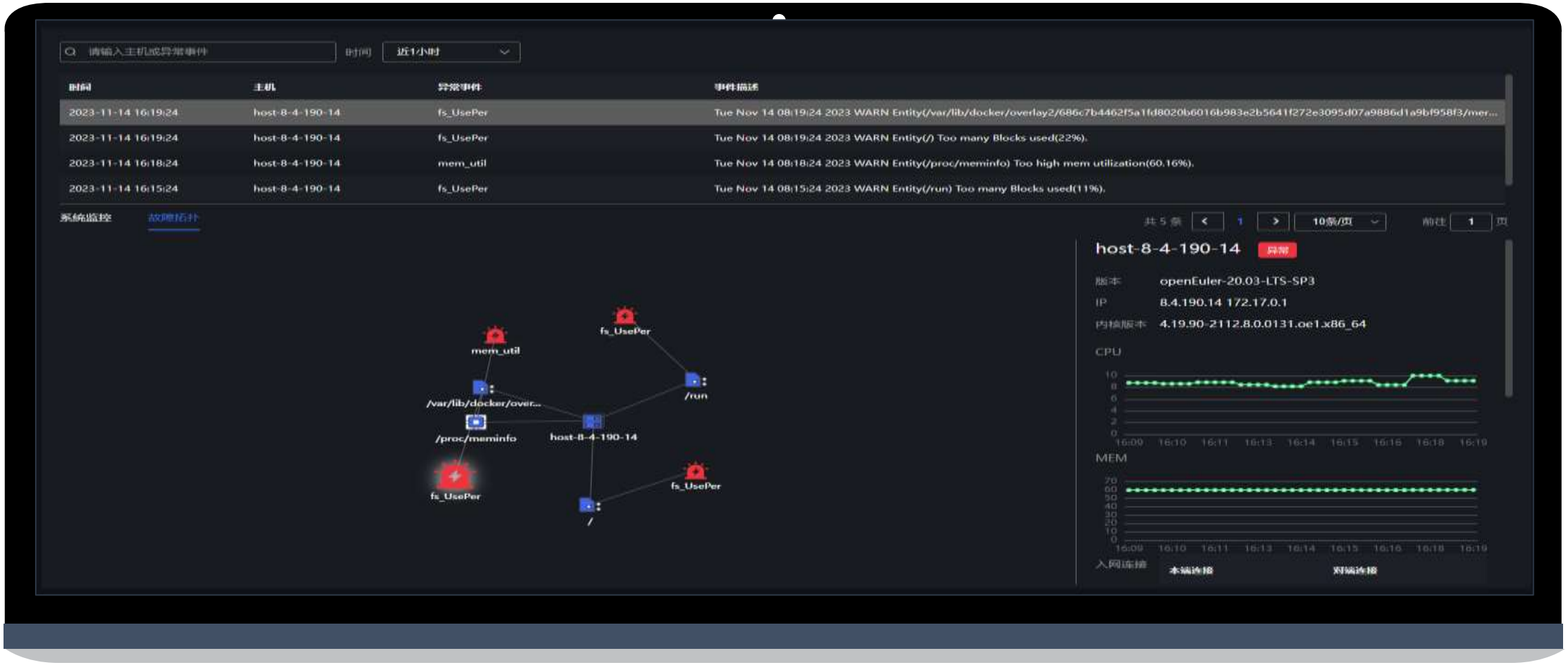
节点级



服务级

- ✓ 定期通过探针采集的所有已安装探针主机的数据，计算它们之间的拓扑关系，提供 OS 级别的拓扑图构建功能。
- ✓ 实时生成动态拓扑，支持自定义拓扑关系对象生成业务拓扑。

系统性能诊断

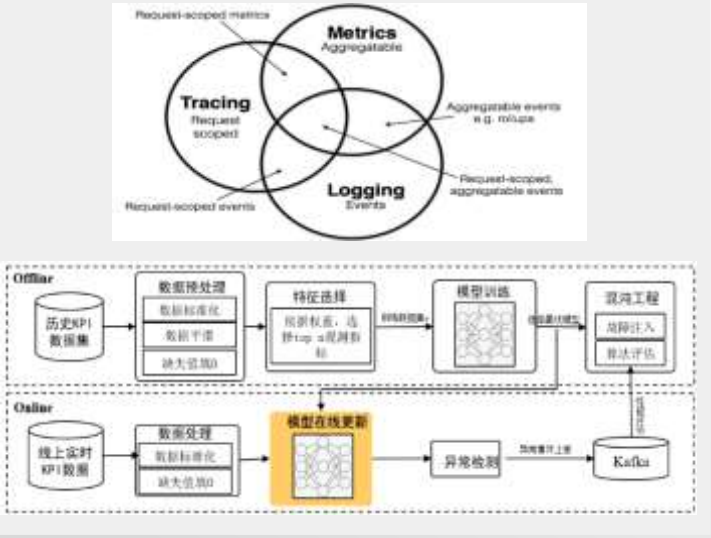


- ✓ 通过eBPF 捕获系统内核数据，通过阈值(包括上下限)设置异常范围，探针会根据阈值判断某个指标是否异常，如果异常则上报异常事件。
- ✓ 系统性能诊断主要用于提供系统维护日常巡检，提供包括如网络、I/O性能波动、Socket队列溢出、系统调用失败、系统调用超时、进程调度超时等。

基于架构感知的系统级故障诊断

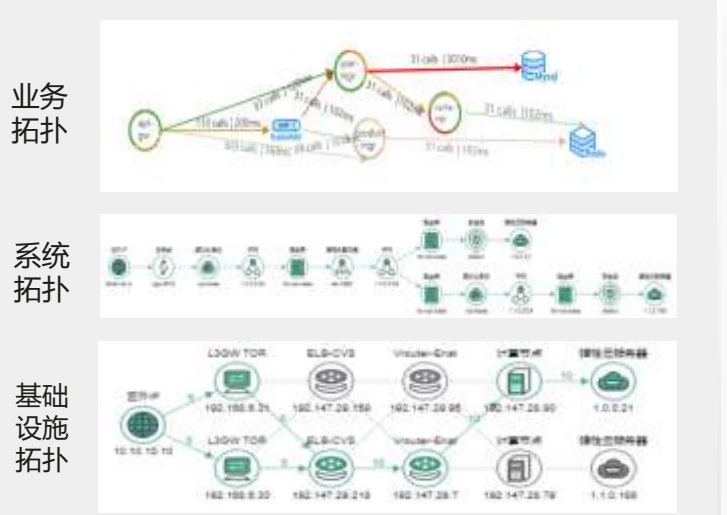
① 异常检测

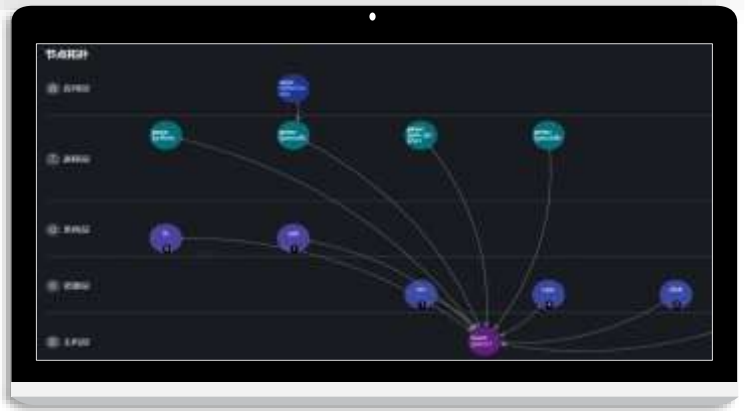
提供应用粒度low-level的数据采集，包括网络、磁盘I/O、调度、内存、安全等方面的系统指标采集，基于采集数据提供分钟级别的异常检测能力，检测发现潜在流量、时延、错误、利用率等业务异常。



② 关联分析

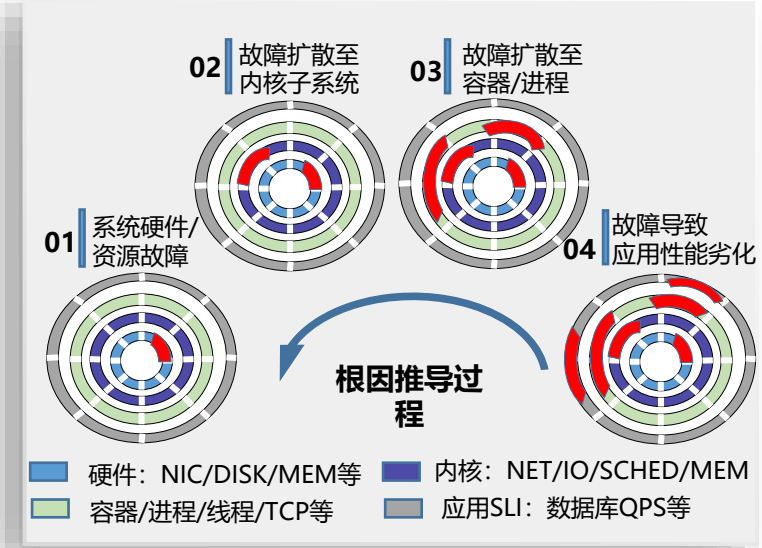
针对系统异常事件采集的系统全栈、全链路数据、业务特征数据，通过关联计算绘制业务拓扑图；结合历史数据模型、日志分析、告警信息、变更分析进行相关分析提取有效诊故障断信息用于最终根因推导。

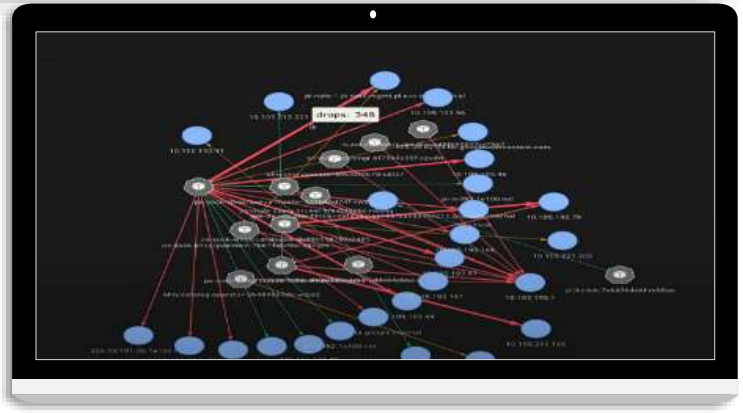




③ 根因推导

基于关联分析的业务拓扑和故障诊断信息，通过定义实体之间的因果关系规则和建立异常事件知识图谱，结合基础组件影响上层应用等运维知识进行因果推导得出根因，实现异常事件根因诊断过程可视化。

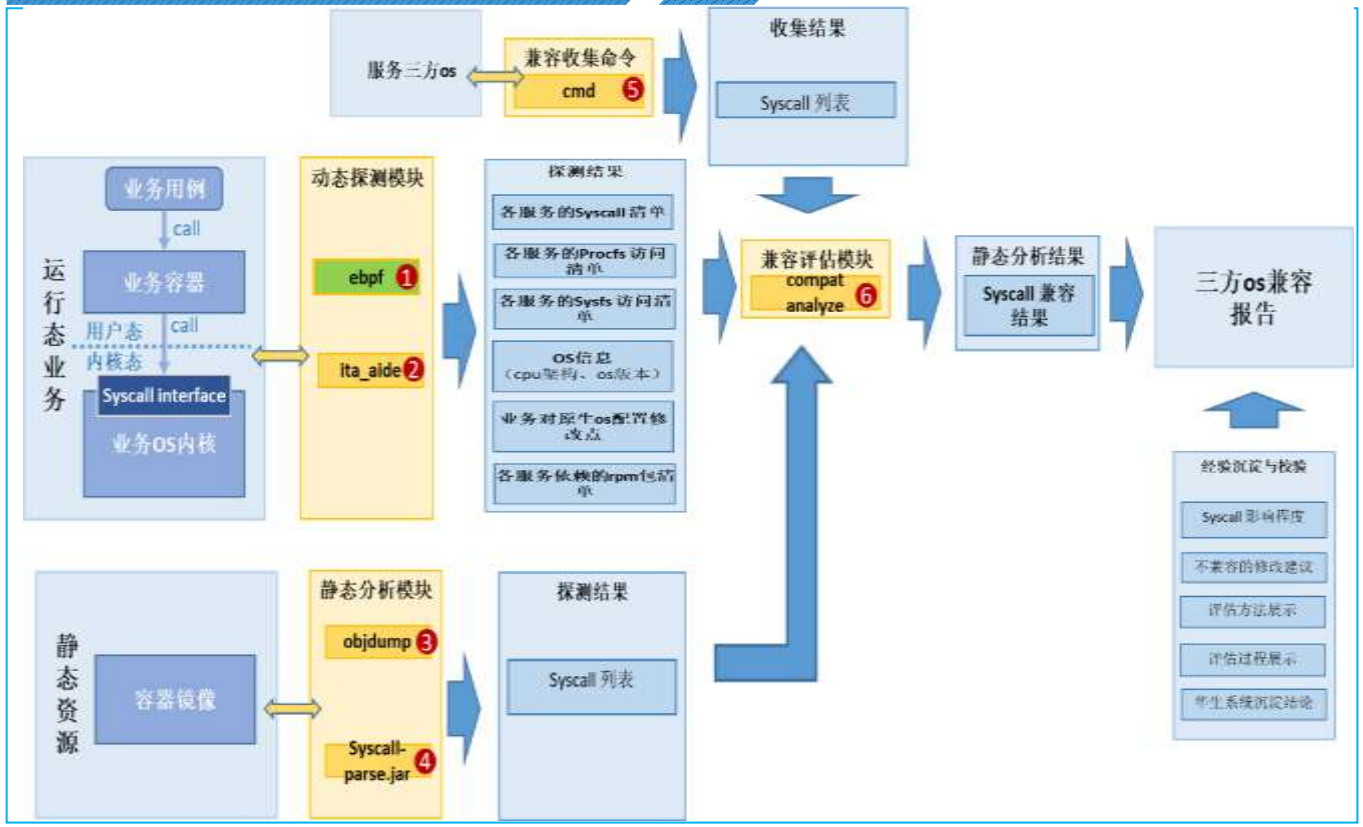




应用实践 – 三方OS兼容性评估工具

通过eBPF动态跟踪系统内核syscall,procfs,sysfs，结合二进制反汇编静态清单，输出系统兼容性分析结果。

关键技术



实践效果

- **支持静态或动态评估兼容性，业务无感知。GDE验证欧拉2.12配套统信、麒麟、bc的评估结果一致。**

容器版本/三方os版本	UnionTech OS Server 20	Kylin Linux Advanced Server V10 (Sword)	BigCloud Enterprise Linux For Euler 21.10 LTS
欧拉2.10 x86	兼容	兼容	兼容
欧拉2.10 aarch64	兼容	兼容	兼容
欧拉2.12 x86	兼容	不兼容，可整改成兼容	不兼容，可整改成兼容
欧拉2.12 aarch64	兼容	条件兼容	条件兼容

- **（三方os数量）*（os版本数量）*（业务数量）*（容器版本数量）的验证工作量**，直接评估GDE应用或者其他业务在任意三方os的兼容性，并提供兼容性报告。

关键技术目标

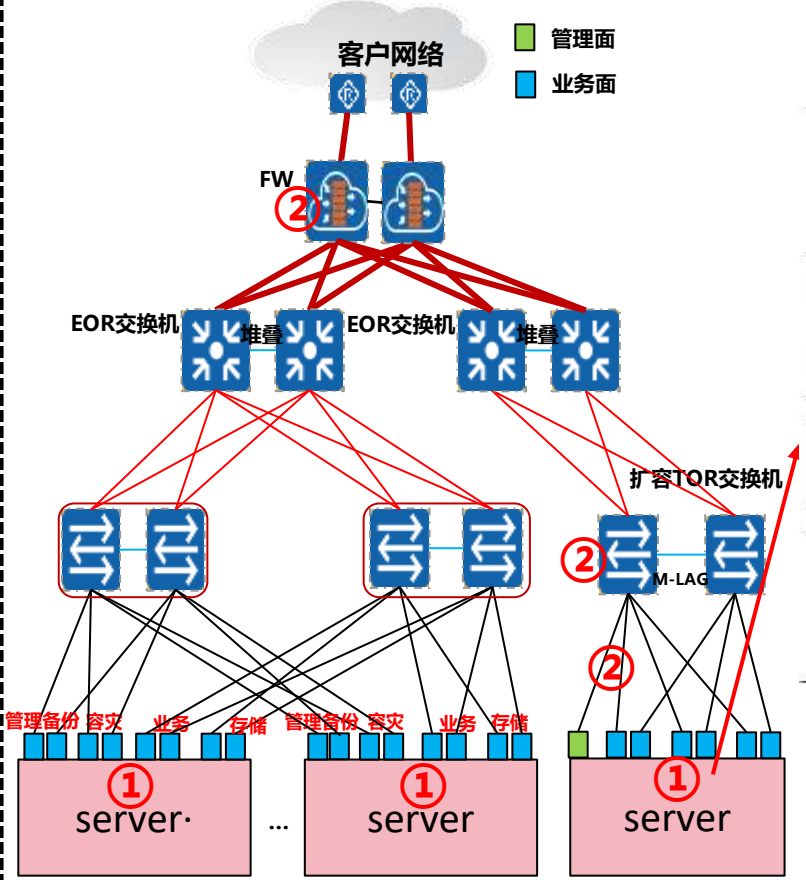
- 帮助业务在任意os上的部署的兼容性分析，降低兼容性分析工作量。

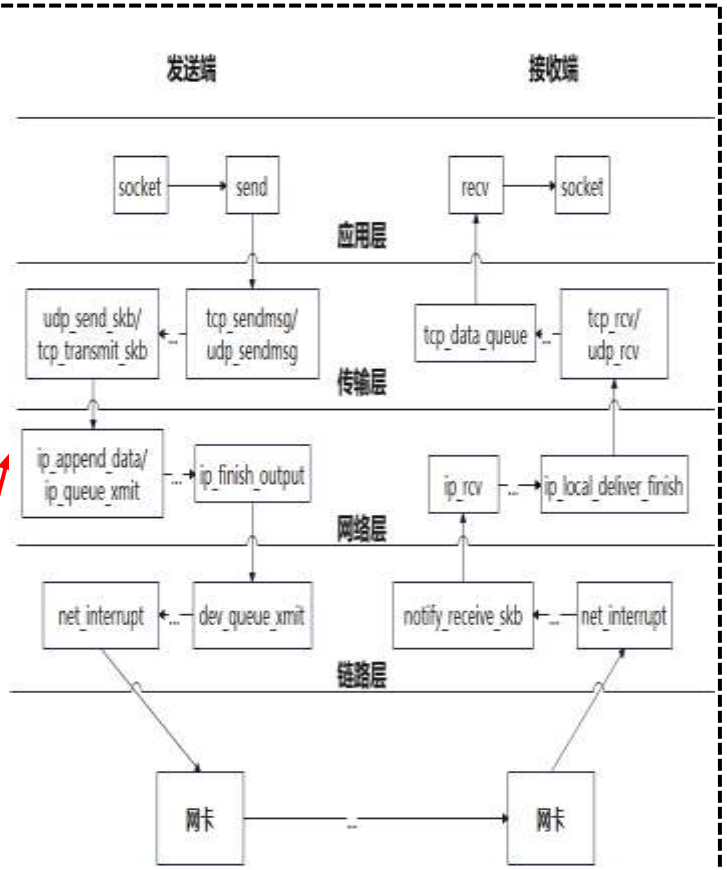
业务价值

低成本的验证业务镜像与目标服务商 OS 是否兼容：

1. 兼容性验证环节业务镜像与服务商 OS 解耦，无需频繁在目标平台部署测试，做到一次探测，全业务受益，在家中环境即可对兼容性进行验证
2. 给出错配场景的可兼容依据，或不兼容的具体原因，避免盲目进行版本配套。

应用实践 - 网络丢包&延时分析工具





适用场景：

① 服务器收发端丢包、延时异常快速定界定位，自证清白

② 中间网络设备、网线侧的问题仍需要通过传统抓包手段进一步定位根因

实现机制：

1、从物理层网卡到应用层所有内核调用函数级追踪，记录端到端延时

2、丢包原因从参考6.10内核的kfree_skb的drop_reason特性，模改到5.10内核，实现相同效果

案例一：防火墙丢包，快速定位原因是被netfilter丢了。

```
[root@host-75-62-70-112 ~]# ./ita-nettrace --drop --addr ^C-btf-path ./vmlinux.btf
[root@host-75-62-70-112 ~]# iptables -t filter -I INPUT 1 -s 75.62.70.110 -p icmp -m icmp --icmp-type
[root@host-75-62-70-112 ~]# iptables-save
# Generated by iptables-save v1.8.7 on Wed Feb 28 11:11:28 2024
*filter
:INPUT ACCEPT [23:1352]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [13:1000]
-A INPUT -s 75.62.70.110/32 -p icmp -m icmp --icmp-type 8 -j DROP
-A INPUT -p icmp -m icmp --icmp-type 13 -j DROP
-A INPUT -p icmp -m icmp --icmp-type 17 -j DROP
-A OUTPUT -p icmp -m icmp --icmp-type 14 -j DROP
-A OUTPUT -p icmp -m icmp --icmp-type 18 -j DROP
COMMIT
# Completed on Wed Feb 28 11:11:28 2024
[root@host-75-62-70-112 ~]# ./ita-nettrace --drop --addr 75.62.70.110 --btf-path ./vmlinux.btf
WARN: skb drop reason is not support by your kernel, will open other probe to analy drop reason.
begin trace...
-------------- kfree_skb happen in nf_hook_slow --------------
Possible causes of packet loss:
        1 : dropped by netfilter
[6363.204269] ICMP: 75.62.70.110 -> 75.62.70.112 ping request, seq: 1, id: 42827

-------------- kfree_skb happen in nf_hook_slow --------------
Possible causes of packet loss:
        1 : dropped by netfilter
[6364.229739] ICMP: 75.62.70.110 -> 75.62.70.112 ping request, seq: 2, id: 42827
```

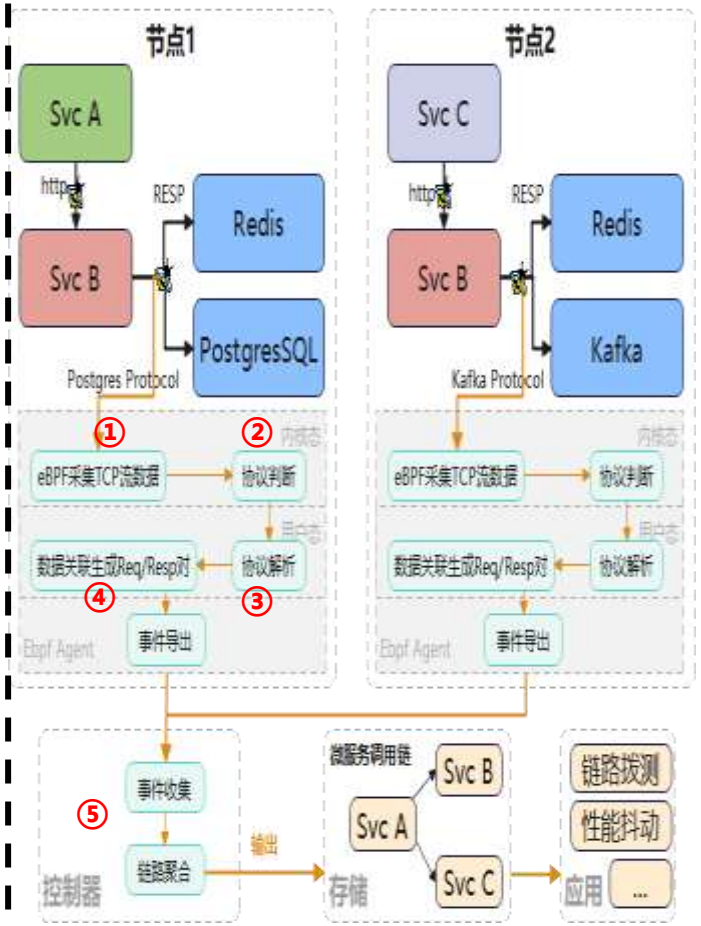
案例二：Traffic Control限流丢包，工具抓取信息直接定位

```
[31401.409644] ICMP: 8.4.67.30 -> 8.4.187.222 ping reply, seq: 16184, id: 61465

-------------- kfree_skb happen in kfree_skb_list --------------
Possible causes of packet loss:
        1 : dropped in TC egress HOOK
```

全链路追踪方案

基于eBPF融合Java agent等技术，实现多语言、全软件栈的观测能力，
支持应用层网络链路的可观测，涵盖常见协议及RED指标



① **数据采集**

基于eBPF采集TCP流数据

② **协议判断**

内核态对tcp流量进行简单分析，判断所属应用层协议，并将数据发送至用户态进行进一步解析。

③ **协议解析**

用户态程序对接受到的tcp流量信息进行解析，根据不同协议提取关键信息并存储。

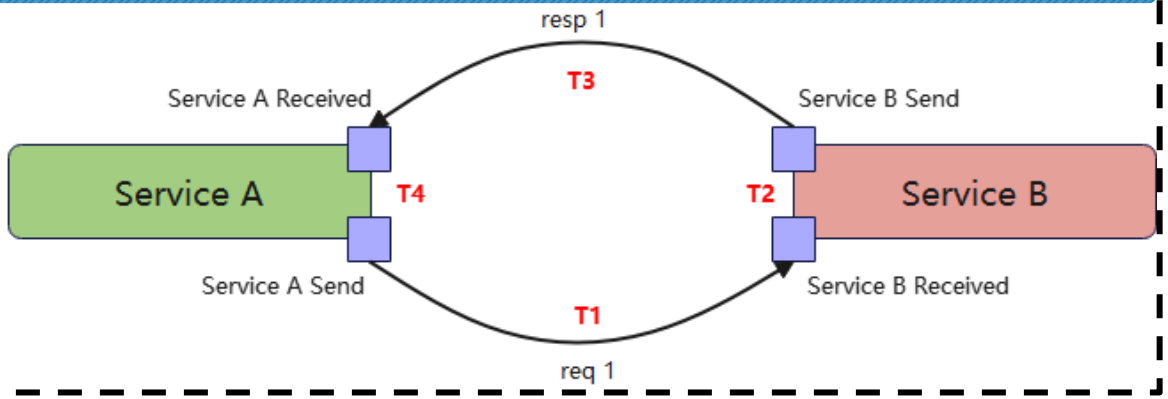
④ **数据匹配**

对步骤3中存储的信息进行匹配，匹配基于**时间戳**和**Thread ID**和**协议报文数据**，将网络请求Req/Resp进行关联重建，并上传至 Prometheus。

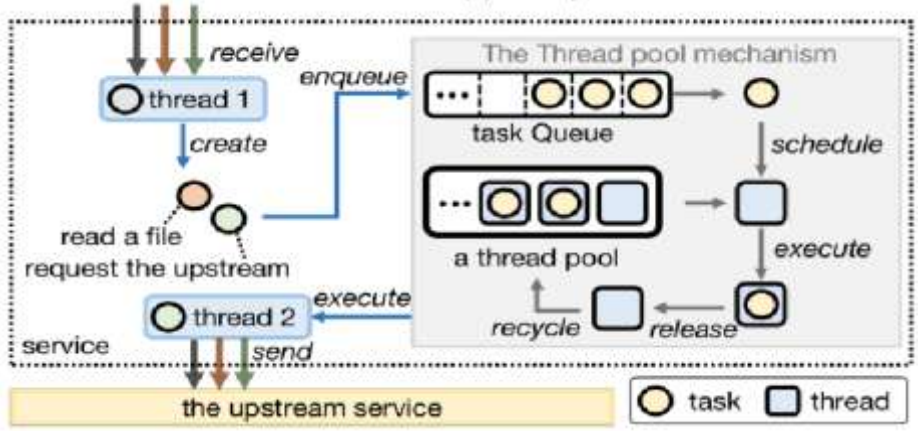
⑤ **链路聚合**

基于步骤4中的网络请求Req/Resp对，通过TCP的序列号进行聚合，得到一条微服务调用链。

T1/T3通过req id关联服务A和B
T2/T4是同一个线程处理，根据时间段进行对应



痛点问题：线程池场景下服务内跨线程receive和send关联中断，如何建立联系



receive如何与send联系起来?

基于eBPF的无侵入式观测技术，打通http 1.X、redis、kafka、mysql、dns等常用协议
实现云原生网络可观测性，支持服务调用路径追踪

应用实践- XXX项目（XXX政务云）

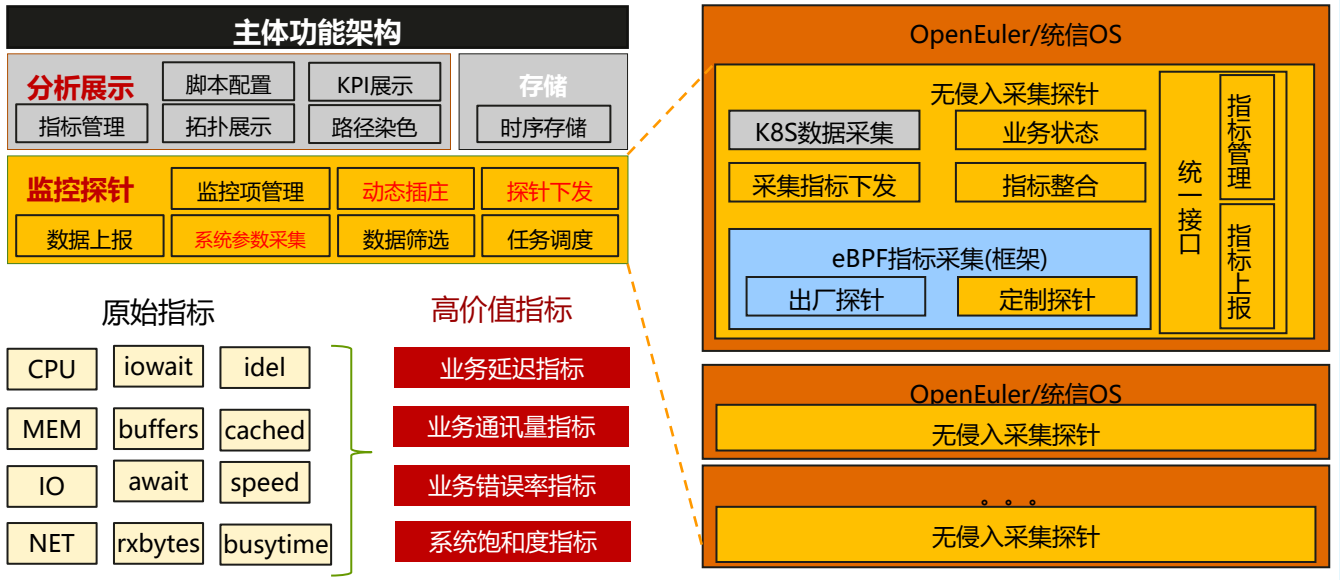
100+指标无侵入秒级采集，应用拓扑还原准确率80%+

现状 应用拓扑手动维护**无法满足客户诉求。**

诉求 基于**无侵入轻量化采集**使能产品应用拓扑自动还原，辅助故障处理。

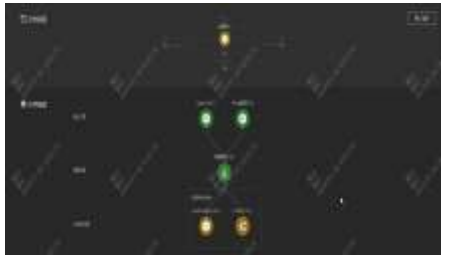
关键技术

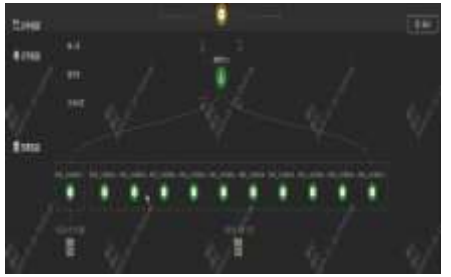
基于**ebpf探针**进行业务诉求的扩展，打造企业运维领域的无侵入低功耗指标采集探针



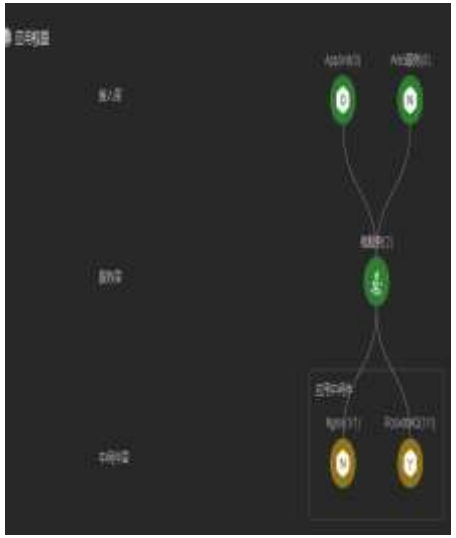
实践效果

应用进程/微服务自动发现





应用组件自动发现



关键技术目标

- POC环境下完成**100+**无侵入指标秒级采集
- 应用调用拓扑还原准确率达到**80%**（基于满足Linux内核要求环境）
- 监听探针功耗不超过单核**CPU 5%**

Thank You