第三届 eBPF开发者大会

www.ebpftravel.com

华为云Stack
基于eBPF的无侵入可观测实践

阙燕文

华为云Stack架构师

中国·西安

华为云Stack（HCS）：通过集中运维、服务治理、部署工具、统一基座构建满足政企云可批量复制、软硬协同的全栈解决方案，持续引领政企云市场

- 连续四年中国软件定义计算(SDC)软件市场份额 **No.1**
- 连续七年中国云系统软件(CSS)市场份额 **No.1**
- 连续四年中国容器软件(CIS)市场份额 **No.1**
- 连续六年中国云系统和服务管理软件市场份额 **No.1**

中国混合云基础架构
领导者

新兴亚太地区混合云领导力
领导者

- 中国大数据平台市场份额 **No.1**
- 中国私有化部署大数据平台市场份额 **No.1**
- 连续八年中国桌面云市场份额 **No.1**
- 中国云专业服务市场份额 **No.1**

政府

NO.1

中国政务云市场份额连续七年
中国数字政府一体化大数据平台市场份额连续三年

900+政务云　**50+**部委　**20+**国家政务云

金融

NO.1

中国金融自建专属云份额年度第一连续六次

300+金融云　**6大行+12股份制银行**

央国企

全量领导者

央国企上云能力服务商

55+央企，3大发电集团、三油一管、三峡集团

Source：IDC、Forrester、沙利文等

HCS管控面可观测问题与挑战

1、物理网络、虚拟网络、k8s集群网络并存，数据流向**错综复杂，治理难度大**；

容器网络	Bridge/Host/Container/serviceMesh …
虚拟网络	VPC/安全组/虚拟交换机/DPDK/VF直通/Vxlan/vlan….
物理网络	Spine-Leaf/BGP路由/VPN……

2、NAT场景、LB、AGW等网关和代理场景带来**流割裂，流还原难度大**；

3、 HCS现网版本10+，可观测方案要**兼容众多版本**带来巨大挑战；

HCS 8.5.0+	HCS 8.3.1	HCS 8.3.0	HCS 8.2.1	HCS 8.2.0	HCS 8.1.1	HCS 8.1.0	HCS 8.0.3	HCS 8.0.2	HCS 8.0.1	HCS 8.0.0

4、HCS作为云平台，在网数量3000+，可观测方案的实施需要对被观测**业务无感**，才有普适性。因而需要**确保无侵入采集**；

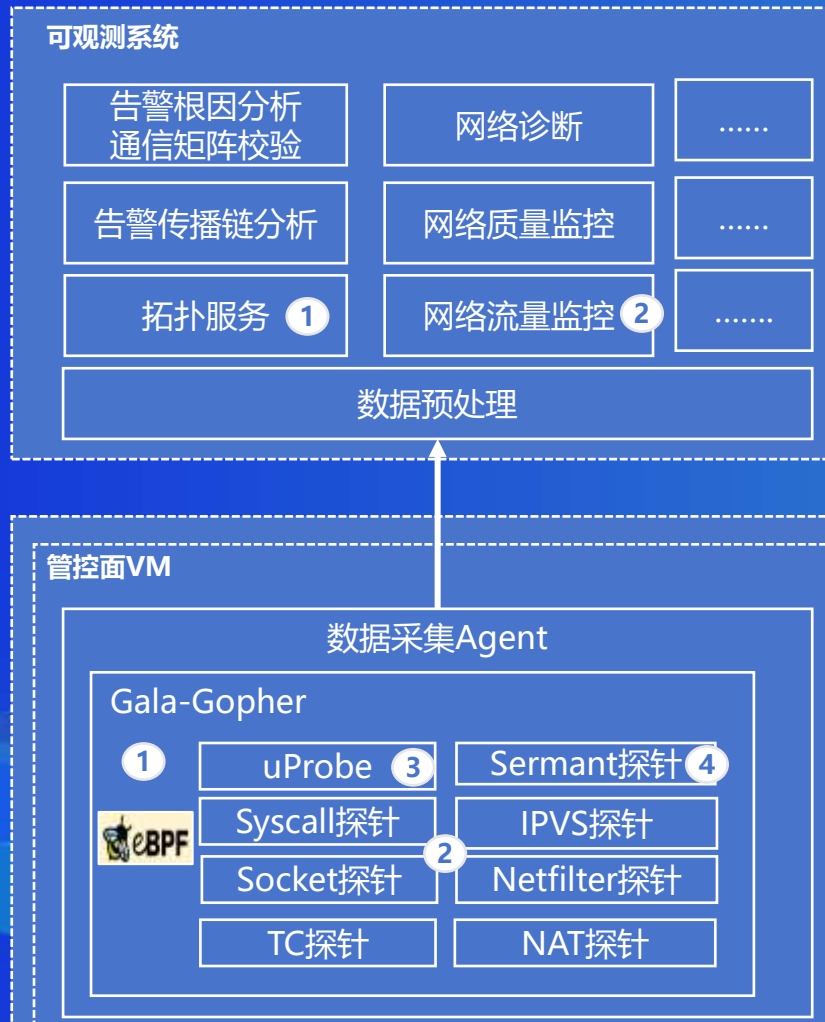
5、HCS拥有14类120+服务，每个服务由不同开发团队负责，侵入式地采集必然带来协作难度和工作量的提升，因而同样需要**无侵入采集**。

6、HCS规模庞大及众多服务的管理，将**产生大量观测数据**，Agent本身的开销以及服务端数据的处理和存储都是一个巨大的挑战。

云服务依赖关系复杂

14类120+云服务

HCS管控面可观测总体方案



第一阶段可观测服务：

① **基于链路的拓扑服务**：通过链路数据构建微服务/进程间的依赖拓扑，精准还原告警传播链，辅助进行**告警根因分析**，辅助**通信矩阵校验**；

② **基于指标的网络诊断**：通过指标数据监控网络链路质量，进行精准的**网络质量实时诊断**。

Agent侧无侵入数据采集：

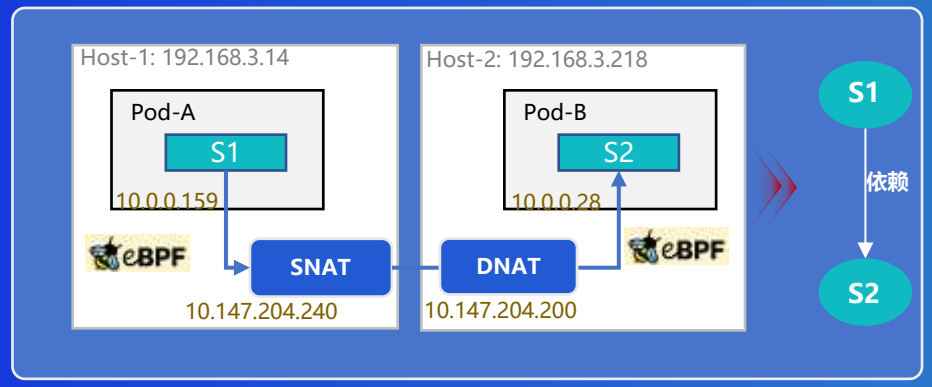
① 采用Gala-Gopher，基于eBPF实现**无侵入**数据采集；

② 在linux内核挂载TC/Socket/Syscall/IPVS/Netfilter等eBPF探针，实现经过内核协议栈的网络链路和网络指标采集，**开发语言无关，普适性强**；

③ 通过eBPF uProbe实现**加密场景、DPDK**等用户空间网络流和指标采集；

④ 通过Sermant补充**JAVA应用场景**应用流和指标采集。

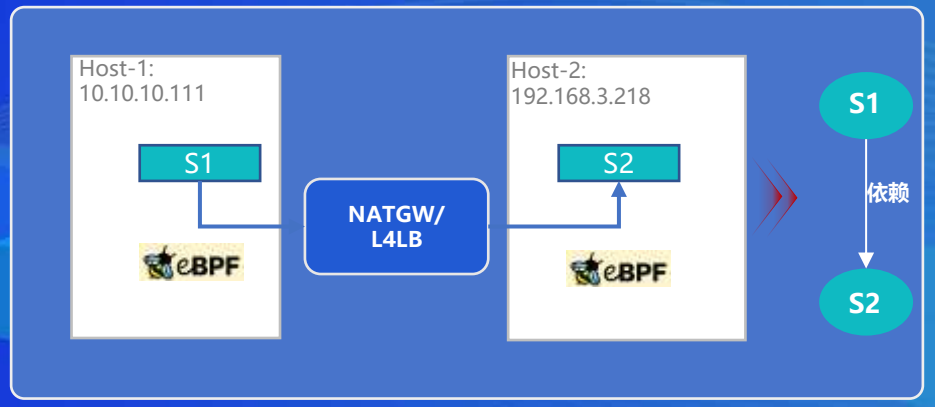
NAT/代理场景链路还原



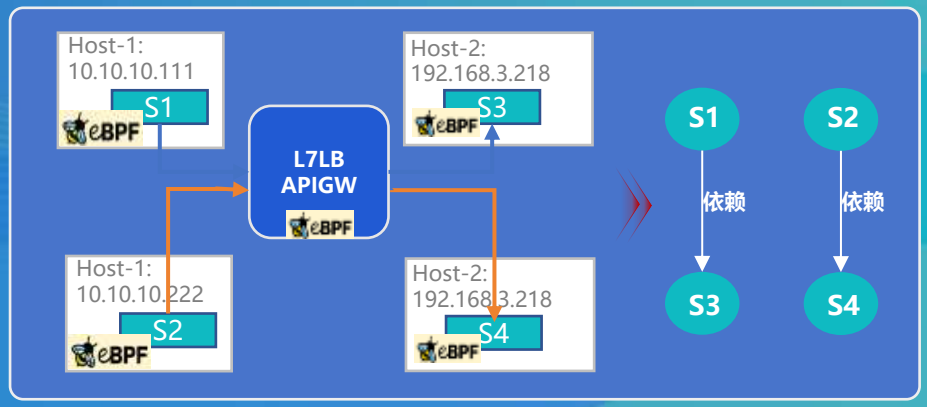
穿越Service网络

通过TCP Options注入，实现各类NAT网关、代理的穿越，真实还原数据链路；

- ≥5.10内核，eBPF Sockops随包注入TCP Options;
- ＜5.10内核，eBPF TC clone数据包，并注入TCP Optons;
- 7层代理场景，通过TOA识别和注入实现转发跟踪;



穿越网关

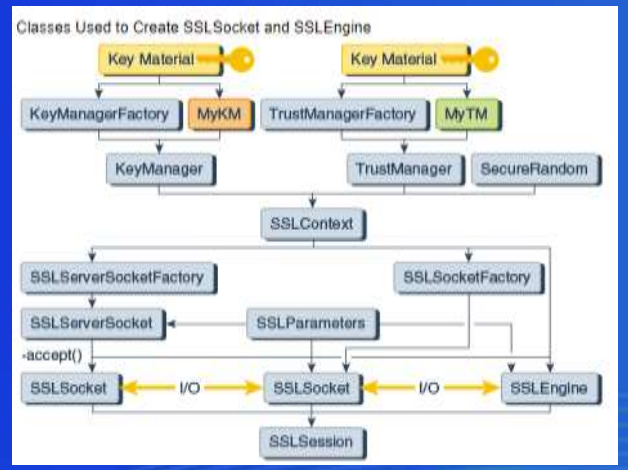


穿越L7代理

eBPF短板及补充方案——sermant

Java应用监控场景短板——uprobe能力受限

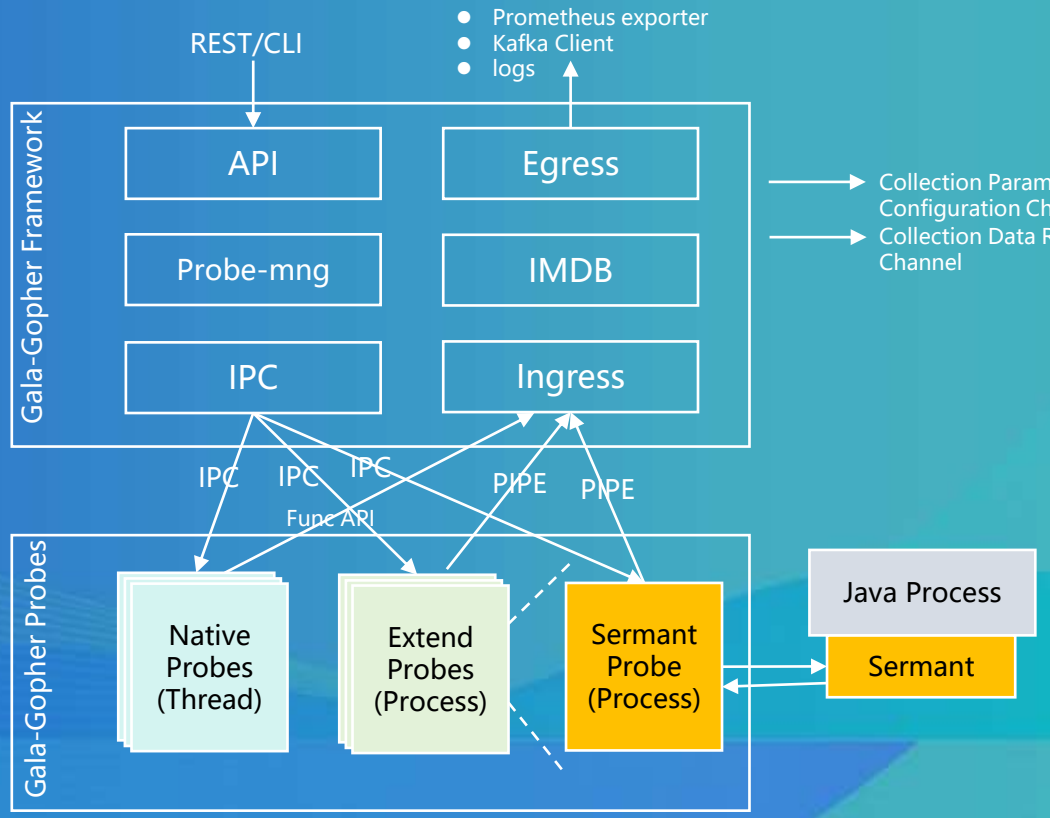
- **场景覆盖不全**
- **Java场景使用eBPF+uprobe成本较高**




```
case NEED_WRAP: {
    socketWriteBuffer.clear();
    SSLEngineResult r = sslEngine.wrap(DUMMY, socketWriteBuffer);
    checkResult(r, true);
    socketWriteBuffer.flip();
    Future<Integer> fwrite = socketChannel.write(socketWriteBuffer);
    fwrite.get();
    break;
}
```

- **案例一**：JSSE提供两种SSL加解密库函数，其中SSLEngine仅提供加解密“工具”，在JSSE中并不维护Socket本身信息
- **案例二**：Java应用场景下gRPC、Dubbo3.0等多种L7应用层指标采集受限...

引入Sermant解决Java场景下的能力不足



在Gala-Gopher框架下扩展一个Sermant探针，引入java agent的能力补充Gala-Gopher在java应用监控场景下的短板

CO-RE/chroot等技术提升OS兼容性

① 通过eBPF的**CO-RE技术**实现一次编译，到处运行；

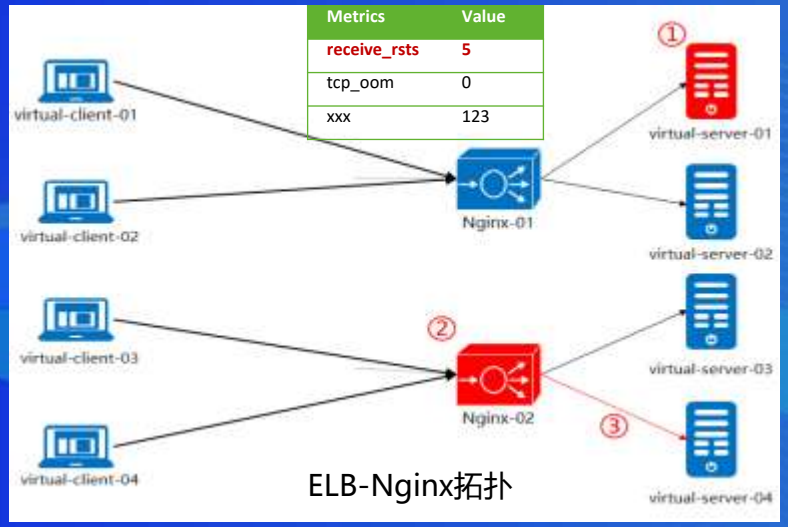
② 通过**chroot实现轻量容器**，隔离Agent运行环境，提升兼容能力。

极致性能


问题
优化方案
效果
可观测系统
数据处理、存储
标签+指标
标签占70%空间
数据上报
带宽占用大
管控面VM
数据采集Agent
eBPF
对Agent、
要求极小开销
• 标签编码存储
• 批量上报+KAE硬加速压缩传输
• 细粒度采集控制
• 自适应采样率
• CPU/内存资源动态申请
• 存储数据量降低到1/3；
• 传输带宽降低到1/15；
• Agent对被观测业务性能无影响；
• Agent极小基础开销：
<CPU 1%， 内存<50M
• 网关类大流量观测场景
Agent CPU开销占业务10%以内

技术方案效果

- HCS管控面核心云服务拓扑准确率达成**99%**；
- HCS云服务通信矩阵测试效率提升：**1周->1天/云服务**；
- ELB应用流跟踪，**提升问题定位/解决效率:**



ELB-Nginx拓扑

序号	问题	原因	eBPF实现ELB应用流采集
①	客户通过7层ELB压测，三万条有几十条报错	【后端业务问题】后端超时配置错误导致回复reset报文。	**取代抓包，偶现故障精准定位：** eBPF可以采集到socket数据中的reset报文，拓扑上指标可直接体现后端业务异常，同时可生成系统告警。
②	APIC 服务异常，客户反馈影响某实时交易的业务	【ELB数据面问题】Nginx进程单核卡死	**关键指标波动回溯查询：** 1. 采集进程CPU占用率可知nginx进程异常； 2. **Nginx和后端服务的数据量减小，时延增大。**
③	客户某业务经过ELB达不到性能要求	【ELB性能问题】后端服务器抓包判断ELB负载合理，最终原因是服务经过云外带宽受限	**流量分布快速理清：** 拓扑可以直接体现Nginx和后端服务器的连接情况和数据量，判断负载均衡是否合理。

未来演进

- 更全面：IaaS/PaaS管控面/数据面**全面实现服务依赖关系治理**；租户面**应用流拓扑/跟踪**；
- 更精准：构建更精准拓扑、告警传播链；**结合LLM智能根因分析**，进一步提升告警根因分析准确率；
- 更高效：结合现有诊断能力，**实现故障自动排查和自恢复**：
  - 异常流量自动拨测，及时诊断网络故障
  - 实时观测与事后拨测相结合，快速定界应用/网络问题
  - 异常流量实时告警，快速通知运维人员定位排障