中期报告

陈思睿　梁恒宇　吕泓涛　汤力宇

中国科学技术大学

2021 年 4 月 26 日

中国科学技术大学
University of Science and Technology of China

sBPF——基于 eBPF 结构的安全沙盒

- 沙盒：一种用于隔离进程的安全机制，用于防御恶意进程和规避系统崩溃。
- eBPF：扩展·伯克利包过滤器（extended Berkeley Packet Filter）—种可以安全的在内核态执行用户代码的框架。

中国科学技术大学
University of Science and Technology of China

eBPF 程序的开发和加载

- 程序的编写：C 或 RUST 直接编写
- 程序的编译：编译器编译成字节码 bytecode
- 程序的加载：用户态程序向 OS 请求加载，verifier 确认安全性，JIT 实时编译成本地机械码，放置特定只读内存段

中国科学技术大学
University of Science and Technology of China

eBPF 程序的运行

- 通过钩子触发程序的执行
- 系统调用通过 helper 函数接口实现
- 局部存储使用 mmap 和对应 helper 实现。

中国科学技术大学
University of Science and Technology of China

优点和缺陷

- 优点：灵活性、安全性、高效率、高兼容性、热升级特性……
  试想一下你要魔改你的内核……
- 缺点：程序的结构、规模和功能类型受限
  程序类型、helper 函数……

中国科学技术大学
University of Science and Technology of China

eBPF 的前景与现状

- eBPF 有着与 JVM 类似的结构，有正在成为流行的通用框架
- 现有的应用：
  - bpftrace 动态监测工具，获取目标进程的所有行为
  - tcpdump 网络包监测工具，提取符合一定条件的所有网络包
  - InKeV 可编程的网络设备负载均衡和路径选择

中国科学技术大学
University of Science and Technology of China

沙盒

- 基本思路是隔离
- 面对可疑进程的潜在威胁并保护自己的系统和数据

中国科学技术大学
University of Science and Technology of China

恶意进程

1. Social Engineering
2. 栈溢出攻击
3. Leak Attack
4. 攻击动态加载器
5. 利用进程列表攻击其他进程
6. 检测沙盒环境并“装死”
7. 窃取隐私文件
8. 强行植入 VNC 远程桌面
9. 僵尸网络

沙盒

1. 局部系统服务
2. 检查内存访问
3. 局部文件系统
4. 设备权限隔离
5. 进程隔离
6. 限制资源使用量
7. 难以感知的监控

eBPF 在沙盒应用的意义

1. BPF 程序能简单的给内核增加新的安全特性并且易于升级
2. 只读代码段不可篡改性不会增加安全漏洞
3. 使用 BPF 可以高效获取资源使用量（现有解决方案）
4. 利用钩子直接劫持系统调用，高效且难以被察觉

中国科学技术大学
University of Science and Technology of China

gVisor 的设计思路

- Ptrace/KVM 两种模式
- 局部文件系统
- Gofer 间接访问本机文件系统
- Seccomp 防止 sentry 被劫持

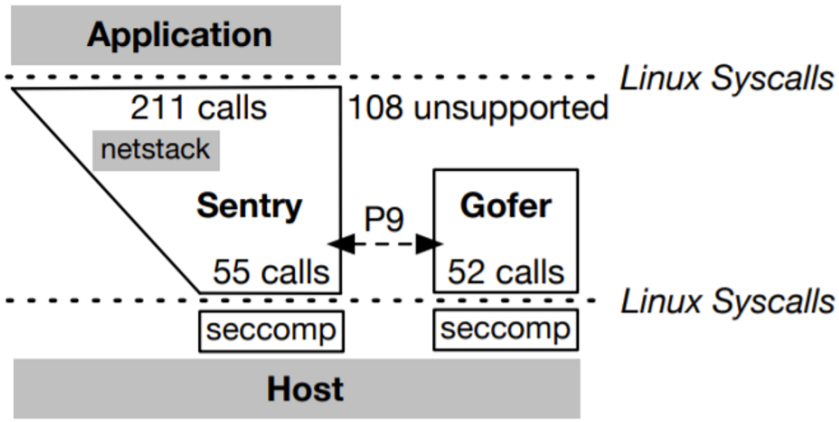
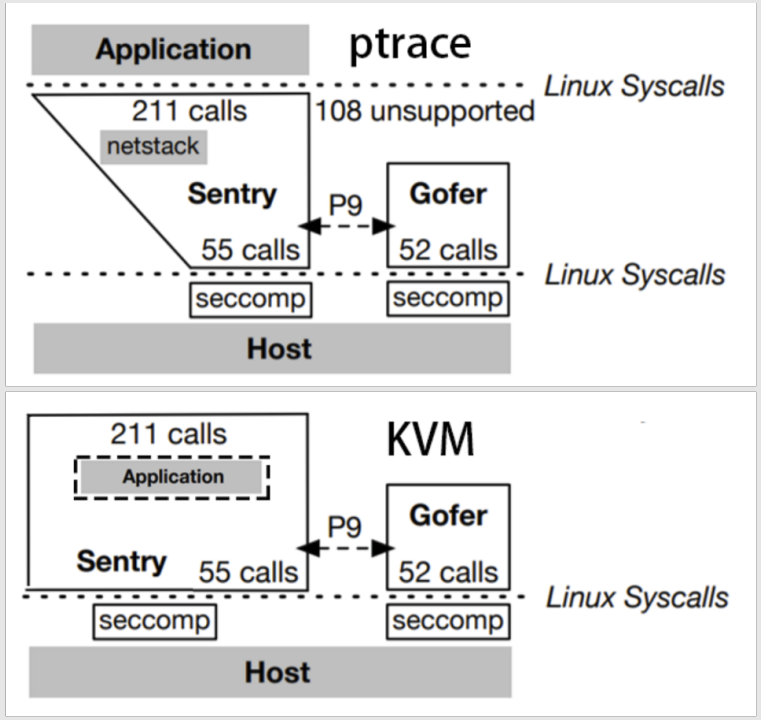


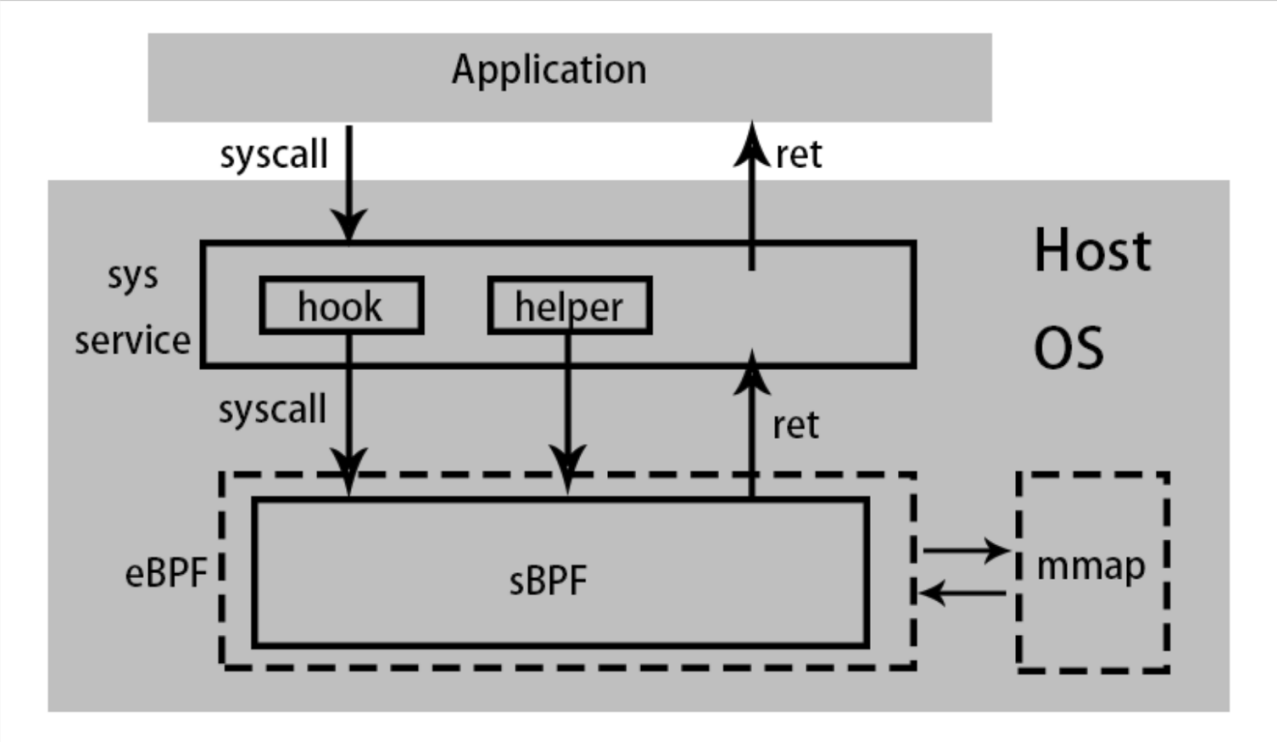
Figure 1: **gVisor Architecture.**

gVisor 的问题

- Ptrace 效率问题?
- Sentry 被污染?
- 虚拟环境被发现?



我们的解决方案——sBPF



我们完成的实践——eBPF 监控新进程创建

```
           <...>-325497  [001] .... 13609.129484: 0: Hello, World!           <..
.>-325498  [002] .... 13609.129564: 0: Hello, World!
           <...>-325499  [003] .... 13610.195734: 0: Hello, World!
           <...>-325500  [000] .... 13610.197371: 0: Hello, World!
           <...>-325501  [002] .... 13610.197623: 0: Hello, World!
           <...>-325502  [002] .... 13611.266022: 0: Hello, World!
           <...>-325503  [003] .... 13611.267731: 0: Hello, World!
           <...>-325504  [000] .... 13611.267816: 0: Hello, World!
           <...>-325505  [001] .... 13612.336308: 0: Hello, World!
           <...>-325506  [002] .... 13612.338039: 0: Hello, World!
           <...>-325507  [003] .... 13612.338171: 0: Hello, World!
           <...>-325508  [001] .... 13613.405017: 0: Hello, World!
           <...>-325509  [003] .... 13613.406619: 0: Hello, World!
           <...>-325510  [000] .... 13613.406740: 0: Hello, World!
           <...>-325511  [003] .... 13614.476889: 0: Hello, World!
           <...>-325512  [000] .... 13614.478580: 0: Hello, World!           <..
.>-325513  [001] .... 13614.478669: 0: Hello, World!
           <...>-325514  [001] .... 13615.598697: 0: Hello, World!
           <...>-325515  [002] .... 13615.600209: 0: Hello, World!
           <...>-325516  [003] .... 13615.600332: 0: Hello, World!
           <...>-325517  [001] .... 13616.668419: 0: Hello, World!
           <...>-325518  [002] .... 13616.670007: 0: Hello, World!
           <...>-325519  [003] .... 13616.670134: 0: Hello, World!
```

中国科学技术大学
University of Science and Technology of China

eBPF 实现的符号跟踪

```
b'            test-14701   [002] ....  3764.278407: 0: New process running with
PID: 14701'
b'            test-14705   [001] ....  3765.257826: 0: New process running with
PID: 14705'
b'            test-14706   [003] ....  3765.874969: 0: New process running with
PID: 14706'
```

图 3: 符号跟踪结果

我们对 even 符号进行跟踪，此时，当且仅当输入偶数时，BPF 才会被触发，显示如下结果：

```
b'            test-16412   [002] ....  4351.014742: 0: New process running with
PID: 16412'
```

图 4: 符号跟踪结果

中国科学技术大学
University of Science and Technology of China

seccomp/BPF 系统调用过滤与拦截

```
1 [pid 319933] write(1, "total 32\ndrwxrwxr-x 2 lhy lhy 40"..., 339) = -1 EPERM (Operation not
      permitted)
2 [pid 319933] close(1)                 = 0
3 [pid 319933] write(2, "ls: ", 4)       = -1 EPERM (Operation not permitted)
4 [pid 319933] write(2, "write error", 11) = -1 EPERM (Operation not permitted)
5 ...
6 [pid 319933] write(2, ": Operation not permitted", 25) = -1 EPERM (Operation not permitted)
7 [pid 319933] write(2, "\n", 1)         = -1 EPERM (Operation not permitted)
```

代码 11: seccomp 程序执行结果

中国科学技术大学
University of Science and Technology of China

规划路线图

	一组	二组
阶段一	eBPF 相关接口设计	沙盒模块化结构设计
阶段二	eBPF 相关接口实现	沙盒接口设计
阶段三	沙盒各模块实现	
阶段四	测试与完善	

中国科学技术大学
University of Science and Technology of China

Q & A

中国科学技术大学
University of Science and Technology of China