



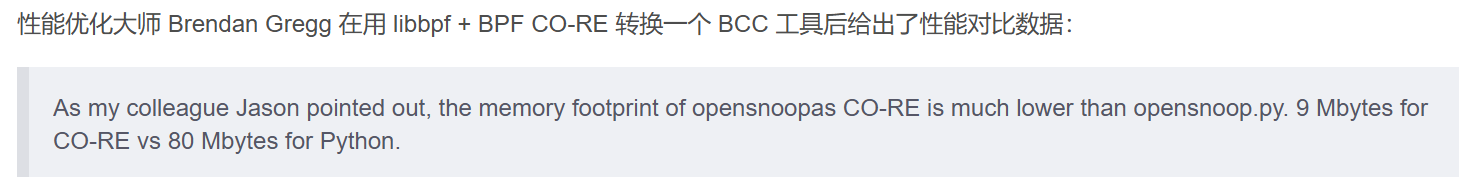




|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 概述 | 语言 | 链接 | 优点 | 缺点 | 灵活性 | 支持功能 | 难度 |  |
| bcc | BPF Compiler Collection, 前端为Python或者Lua,后端为C. 依赖 | 后端C前端Python Lua | 19k | 1.丰富的功能2.易于使用 | 1.学习曲线陡峭，需要学习BCC API  2.BCC运行时依赖Clang等编译工具 | 高 | 提供100+功能 | 使用难度:低学习难度：高 |  |
| libbpf | 支持BPF CO-RE的基础库，是eunomia-bpf的基础 | 后端C前端C | 1.8K | 1.灵活性高 | 1.较低的抽象程度2.学习曲线陡峭，需要学习底层逻辑 | 超高 | 无 | 使用难度:超高学习难度：一般 |  |
| bpftrace | 基于BCC的高层次tracing语言，提供高级语法实现ebpf功能开发 | 后端C前端bpftrace | 7.7k | 1.简单易用的语法 | 1.需要学习特定的语法2.功能相对有限，定制化功能难以实现 | 一般 | 提供40+功能 | 使用难度:中学习难度：一般 |  |
| eunomia-bpf | 基于libbpf的易用框架，开发者只用关心内核态程序开发，用户态程序由框架提供 | 后端C前端Rust C | 500+ | 1.简单易上手，有中文资料2.灵活性中等 |  | 一般 | 提供10+功能 | 使用难度:中学习难度：低 |  |
| ebpf-go | 纯Go实现的ebpf框架 | 后端Go前端Go | 5.5k | 1.纯Go，适合Go应用 | 1.需要学习Go语法和特有API | 高 |  | 使用难度:中学习难度：中 |  |

[HOWTO： BCC 到 libbpf 的转换 ·BPF](https://facebookmicrosites.github.io/bpf/blog/2020/02/20/bcc-to-libbpf-howto-guide.html)

[eBPF系列学习（4）了解libbpf、CO-RE (Compile Once – Run Everywhe） | 使用go开发ebpf程序（云原生利器cilium ebpf ）\_ebpf co-re-CSDN博客](https://docker.blog.csdn.net/article/details/125170099)

[Building BPF applications with libbpf-bootstrap](https://nakryiko.com/posts/libbpf-bootstrap/)

**B站**

**张口就问**

**dave\_di**

**BCC:runqslower，查看被谁抢占和精准的调度延迟**

**Bcc很多内容都迁移到了libbpf**

