



链滴

我写了个假的 Gitlab Runner，可以运行自己的测试和编译自己，在每个星期四运行 job 还会有特殊效果（狗头）

作者：nanmu42

原文链接：<https://ld246.com/article/1666626843152>

来源网站：链滴

许可协议：[署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

Github 仓库

<https://github.com/nanmu42/tart>

摘要

蛋挞是一个教学目的，非官方的 Gitlab Runner，通过简明地实现 Gitlab Runner 功能的一个子集，示 Gitlab Runner 的设计和实现方法。

举个例子，蛋挞可以运行自己的 CI job，运行自己的测试和编译自己。

特色：

- 折腾；
- 使用 [Firecracker](#)和 [/dev/kvm](#)，让每个 job 在一个**两秒内**启动的虚拟机中运行，我目前没在公开料里查到这么做的；
- 代码量少，大概 2000 行实现了 Gitlab Runner 的核心功能：job 的获取、执行、环境隔离、日志结果的上报；
- 在每个星期四运行 job 会有特殊效果。

只实现了核心功能，产物上传、service 这些功能是不支持的。换句话说，不要用于生产环境（真的有人这么做吗）。

使用方法

蛋挞需要在可以访问 [/dev/kvm](#)的 Linux 环境下运行：

```
sudo setfacl -m u:${USER}:rw /dev/kvm
```

1. 从 [release](#) 页面下载蛋挞和 Firecracker 的二进制，并将它们置于 [\\$PATH](#)
2. 从 [release](#) 页面下载 RootFS 和 Linux 内核，把它们放到工作文件夹，比如 [~/tart](#)
3. 为 tart 创建的虚拟机预先配置网络，请参考 [rootfs/setup-tuntap.sh](#)
4. cd 到工作文件夹
5. 注册 tart 为你项目的 Gitlab Runner：[tart register --endpoint https://gitlab.example.com --oken your_token_here > tart.toml](#)
6. 启动 tart：[tart run](#)
7. 在 Gitlab 上触发 CI，为了确保 job 会调度到 tart 上，你可能得禁用项目的 shared runner
8. 观看 tart 工作（或者爆炸）

编译方式

```
make
```

产物在 [bin](#)文件夹中。

虚拟机的 RootFS 和 Linux 内核编译请参考 [rootfs](#)文件夹。

为啥叫蛋挞？

我喜欢吃蛋挞。

许可证

MIT

请自由享受和贡献开源。

第三方项目许可证请参阅 [THIRD_PARTY_LICENSES.md](#).

logo 的照片来自于 Ashley Byrd on Unsplash , Gopher 在[gopherize.me](#)生成。