


 This repository

[Pull requests](#) [Issues](#) [Gist](#)

cyfdecyf / cow

Watch

329

★ Unstar

4,207

Fork

906

<> Code

🔔 Issues

168

🔗 Pull requests

13

📖 Wiki

📶 Pulse

📊 Graphs

HTTP proxy written in Go. COW can automatically identify blocked sites and use parent proxies to access.

📦 1,059 commits

🌿 2 branches

📦 32 releases

👤 15 contributors

Branch: master

New pull request

Create new file

Upload files

Find file










Clone or download

 cyfdecyf Merge branch 'develop', version 0.9.8

Latest commit 41c0fb1 on 19 Jun

doc	Bump version to 0.9.7	4 months ago
script	Fix PAC error in test script.	a year ago
testdata	Add test for isFileExists.	4 years ago
.gitignore	Bump version to 0.9.7	4 months ago
.travis.yml	Use Go 1.4.2 on travis.	a year ago
CHANGELOG	Bump version to 0.9.8	3 months ago
LICENSE	Update CHANGELOG, LICENSE, README.	4 years ago
README-en.md	Merge branch 'develop', version 0.9.8	3 months ago
README.md	Merge branch 'develop', version 0.9.8	3 months ago
auth.go	Allow client use basic authentication.	3 years ago
auth_test.go	Check port in authentication.	3 years ago
config.go	Bump version to 0.9.8	3 months ago
config_test.go	Implement #328, allow specifying stat/blocked etc. in config file.	a year ago
config_unix.go	Implement #328, allow specifying stat/blocked etc. in config file.	a year ago
config_windows.go	Implement #328, allow specifying stat/blocked etc. in config file.	a year ago
conn_pool.go	Close idle upon too many open fd error.	2 years ago
conn_pool_test.go	Fix conn_pool test, not covering multiplex channel.	3 years ago
error.go	Show host name in error page.	3 years ago
estimate_timeout.go	Remove dependency on global config in estimateTimeout().	a year ago
http.go	Fixed wechat image url bug.	3 months ago
http_test.go	Don't use bytes.SplitN in parseRequestURIBytes()	3 years ago
install-cow.sh	Bump version to 0.9.8	3 months ago
log.go	Setup all logger before calling initLog().	a year ago
main.go	Fix cross compile for Windows because no SIGUSR1.	a year ago
main_unix.go	Fix cross compile for Windows because no SIGUSR1.	a year ago
main_windows.go	Fix cross compile for Windows because no SIGUSR1.	a year ago
pac.go	解决微信朋友圈图片无法打开的问题	3 months ago
pac.js	Fix #292: return direct for *.local host in PAC.	a year ago
parent_proxy.go	Fix typo.	a year ago
proxy.go	wait for unfinished jobs and close all accepts	a year ago
proxy_test.go	Refactor proxy server representation.	3 years ago
proxy_unix.go	Fix conn reset handling for go 1.6.1	3 months ago
proxy_windows.go	Close idle upon too many open fd error.	2 years ago

2016年9月8日cyfdecyf/cow: HTTP proxy written in Go. COW can automatically identify blocked sites and use parent proxies to access.

 site_blocked.go	Temp blocked should increase blocked visit count once.	4 years ago
 site_direct.go	Remove MS sites in direct list.	2 years ago
 sitestat.go	Fix PAC error in test script.	a year ago
 sitestat_test.go	Fix test as new site is consider as direct now.	3 years ago
 ssh.go	Support multiple sshServer options.	3 years ago
 stat.go	Collect client&server connection count, better debug message.	3 years ago
 timeoutset.go	Abstract data structure TimeoutSet from chouTime.	4 years ago
 util.go	Remove unused function copyNWithBuf.	a year ago
 util_test.go	Remove unused function copyNWithBuf.	a year ago

 README.md

COW (Climb Over the Wall) proxy

COW 是一个简化穿墙的 HTTP 代理服务器。它能自动检测被墙网站，仅对这些网站使用二级代理。

[English README](#).

当前版本: 0.9.8 [CHANGELOG](#) build passing

欢迎在 [develop branch](#) 进行开发并发送 [pull request](#) :)

功能

COW 的设计目标是自动化，理想情况下用户无需关心哪些网站无法访问，可直连网站也不会因为使用二级代理而降低访问速度。

- 作为 HTTP 代理，可提供给移动设备使用；若部署在国内服务器上，可作为 APN 代理
- 支持 HTTP, SOCKS5, [shadowsocks](#) 和 cow 自身作为二级代理
 - 可使用多个二级代理，支持简单的负载均衡
- 自动检测网站是否被墙，仅对被墙网站使用二级代理
- 自动生成包含直连网站的 PAC，访问这些网站时可绕过 COW
 - 内置 [常见可直连网站](#)，如国内社交、视频、银行、电商等网站（可手工添加）

快速开始

安装:

- OS X, Linux (x86, ARM): 执行以下命令（也可用于更新）

```
curl -L git.io/cow | bash
```

 - 环境变量 `COW_INSTALLDIR` 可以指定安装的路径，若该环境变量不是目录则询问用户
 - 所有 binary 在 OS X 上编译获得，若 ARM 版本可能无法工作，请下载 [Go ARM](#) 后从源码安装
- Windows: 从 [release 页面](#) 下载
- 熟悉 Go 的用户可用 `go get github.com/cyfdecyf/cow` 从源码安装

编辑 `~/.cow/rc` (Linux) 或 `rc.txt` (Windows)，简单的配置例子如下:

```
#开头的行是注释，会被忽略
# 本地 HTTP 代理地址
# 配置 HTTP 和 HTTPS 代理时请填入该地址
# 若配置代理时有对所有协议使用该代理的选项，且你不清楚此选项的含义，请勾选
# 或者在自动代理配置中填入 http://127.0.0.1:7777/pac
listen = http://127.0.0.1:7777

# SOCKS5 二级代理
```

```
proxy = socks5://127.0.0.1:1080
# HTTP 二级代理
proxy = http://127.0.0.1:8080
proxy = http://user:password@127.0.0.1:8080
# shadowsocks 二级代理
proxy = ss://aes-128-cfb:password@1.2.3.4:8388
# cow 二级代理
proxy = cow://aes-128-cfb:password@1.2.3.4:8388
```

使用 cow 协议的二级代理需要在海外服务器上安装 COW，并使用如下配置：

```
listen = cow://aes-128-cfb:password@0.0.0.0:8388
```

完成配置后启动 COW 并配置好代理即可使用。

详细使用说明

配置文件在 Unix 系统上为 `~/.cow/rc`，Windows 上为 COW 所在目录的 `rc.txt` 文件。[样例配置](#) 包含了所有选项以及详细的说明，建议下载然后修改。

启动 COW：

- Unix 系统在命令行上执行 `cow &` (若 COW 不在 `PATH` 所在目录，请执行 `./cow &`)
 - [Linux 启动脚本](#)，如何使用请参考注释（Debian 测试通过，其他 Linux 发行版应该也可使用）
- Windows
 - 双击 `cow-taskbar.exe`，隐藏到托盘执行
 - 双击 `cow-hide.exe`，隐藏为后台程序执行
 - 以上两者都会启动 `cow.exe`

PAC url 为 `http://<listen address>/pac`，也可将浏览器的 HTTP/HTTPS 代理设置为 `listen address` 使所有网站都通过 COW 访问。

使用 PAC 可获得更好的性能，但若 PAC 中某网站从直连变成被封，浏览器会依然尝试直连。遇到这种情况可以暂时不使用 PAC 而总是走 HTTP 代理，让 COW 学习到新的被封网站。

命令行选项可以覆盖部分配置文件中的选项、打开 `debug/request/reply` 日志，执行 `cow -h` 来获取更多信息。

手动指定被墙和直连网站

一般情况下无需手工指定被墙和直连网站，该功能只是是为了处理特殊情况和性能优化。

配置文件所在目录下的 `blocked` 和 `direct` 可指定被墙和直连网站（`direct` 中的 `host` 会添加到 PAC）。Windows 下文件名为 `blocked.txt` 和 `direct.txt`。

- 每行一个域名或者主机名（COW 会先检查主机名是否在列表中，再检查域名）
 - 二级域名如 `google.com` 相当于 `*.google.com`
 - `com.hk`，`edu.cn` 等二级域名下的三级域名，作为二级域名处理。如 `google.com.hk` 相当于 `*.google.com.hk`
 - 其他三级及以上域名/主机名做精确匹配，例如 `plus.google.com`

技术细节

访问网站记录

COW 在配置文件所在目录下的 `stat` `json` 文件中记录经常访问网站被墙和直连访问的次数。

- 对未知网站，先尝试直接连接，失败后使用二级代理重试请求，2 分钟后再尝试直接
 - 内置[常见被墙网站](#)，减少检测被墙所需时间（可手工添加）
- 直连访问成功一定次数后相应的 `host` 会添加到 PAC
- `host` 被墙一定次数后会直接用二级代理访问
 - 为避免误判，会以一定概率再次尝试直连访问
- `host` 若一段时间没有访问会自动被删除（避免 `stat` 文件无限增长）

- 内置网站列表和用户指定的网站不会出现在统计文件中

COW 如何检测被墙网站

COW 将以下错误认为是墙在作怪:

- 服务器连接被重置 (connection reset)
- 创建连接超时
- 服务器读操作超时

无论是普通的 HTTP GET 等请求还是 CONNECT 请求, 失败后 COW 都会自动重试请求。(如果已经有内容发送回 client 则不会重试而是直接断开连接。)

用连接被重置来判断被墙通常来说比较可靠, 超时则不可靠。COW 每隔半分钟会尝试估算合适的超时时间间隔, 避免在网络连接差的情况下把直连网站由于超时也当成被墙。COW 默认配置下检测到被墙后, 过两分钟再次尝试直连也是为了避免误判。

如果超时自动重试给你造成了问题, 请参考[样例配置](#)高级选项中的 `readTimeout`, `dialTimeout` 选项。

限制

- 不提供 cache
- 不支持 HTTP pipeline (Chrome, Firefox 默认都没开启 pipeline, 支持这个功能容易增加问题而好处并不明显)

致谢 (Acknowledgements)

贡献代码:

- @fzerorubigd: various bug fixes and feature implementation
- @tevino: http parent proxy basic authentication
- @xupefei: 提供 cow-hide.exe 以在 windows 上在后台执行 cow.exe
- @sunteya: 改进启动和安装脚本

Bug reporter:

- GitHub users: glacjay, trawor, Blaskyy, lucifer9, zellux, xream, hieixu, fantasticfears, perrywky, JayXon, graminc, WingGao, polong, dallasca, luosheng
- Twitter users: 特别感谢 @shao222 多次帮助测试新版并报告了不少 bug, @xixitalk

@glacjay 对 0.3 版本的 COW 提出了让它更加自动化的建议, 使我重新考虑 COW 的设计目标并且改进成 0.5 版本之后的工作方式。

