**Grafana编译简略**

Go语言配置略过，注意需要配置GOPATH，运行一个Hello World…

# Grafana编译准备：

源码：<https://github.com/grafana/grafana>

复制整个grafana文件夹至%GOPATH%/src/github.com/grafana/文件夹下，注意：红色的路，径需要自己手动创建， 或者直接运行命令go get github.com/grafana/grafana

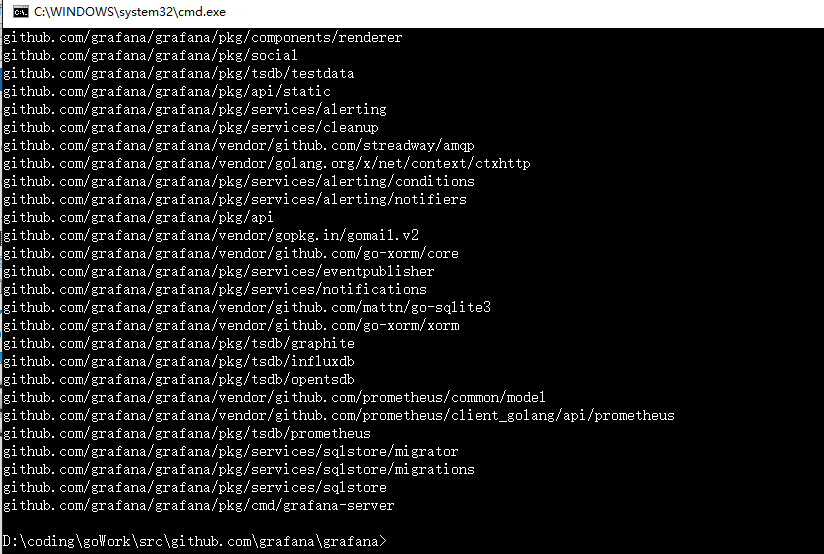
使用公司git私服下载：



# 开始编译

注意：以下命令均在%GOPATH%/src/github.com/grafana/grafana目录中运行

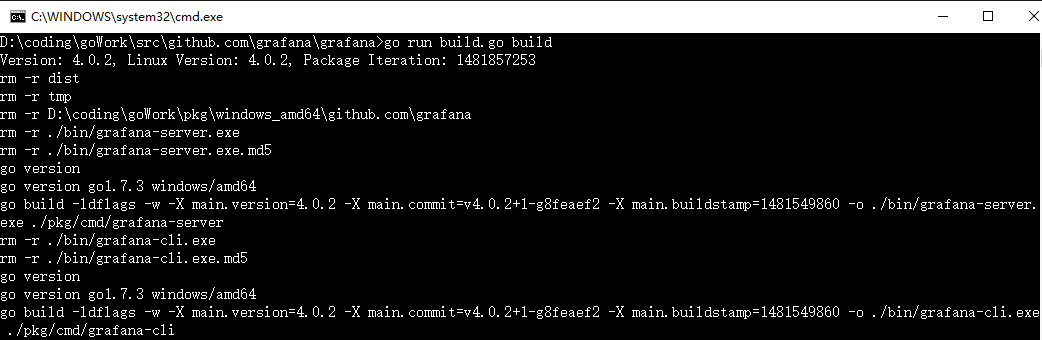
## go run build.go setup



第一次需要下载github.com/kardianos/govendor，可能需要翻墙

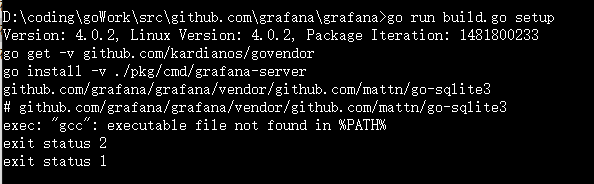


## go run build.go build



# GO可能的错误

## 错误一：GCC编译器的环境



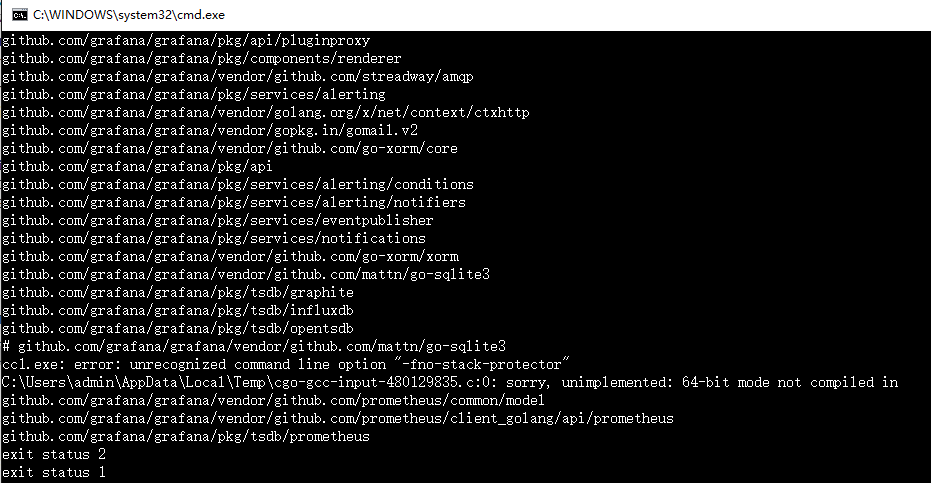


需要下载mingw-w64，32位的不好使，安装完成将bin文件夹添置PATH，下载页面：

https://sourceforge.net/projects/mingw-w64/files/

https://sourceforge.net/projects/mingw-w64/files/Toolchains%20targetting%20Win64/Personal%20Builds/mingw-builds/6.2.0/threads-posix/seh/x86\_64-6.2.0-release-posix-seh-rt\_v5-rev1.7z

## 错误二：go-sqlite3 出现问题



Go 第三方包 go-sqlite3 出现问题，手动安装

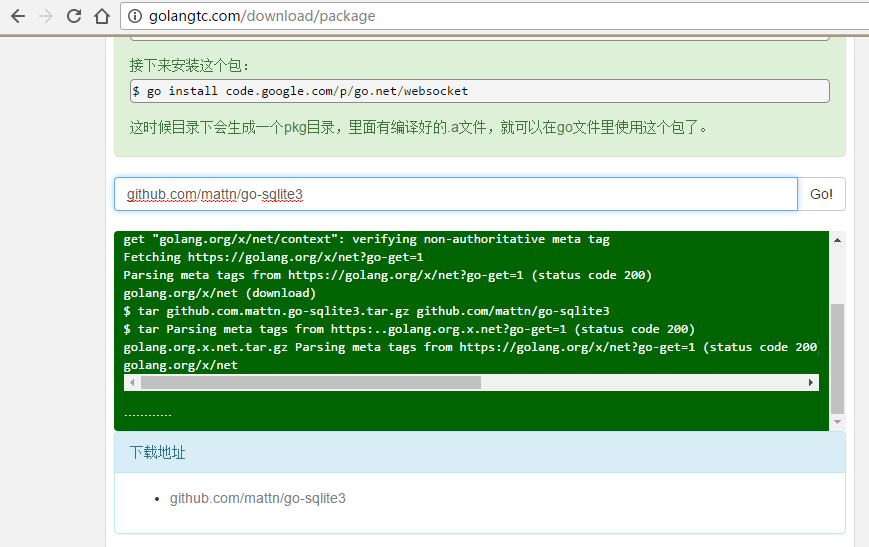
下载地址：<http://www.sqlite.org/2016/sqlite-dll-win64-x64-3150200.zip>，将sqlite3.dll 复制至文件夹C:\Windows\System32下，运行命令go get github.com/mattn/go-sqlite3

（经后期安装MinGW， 才了解不是sqlliet包问题，起始一直是使用32的Gcc打包64位的sqllie，所以以下非翻墙方式获取sqllite包可以忽略）

Gopm下载：



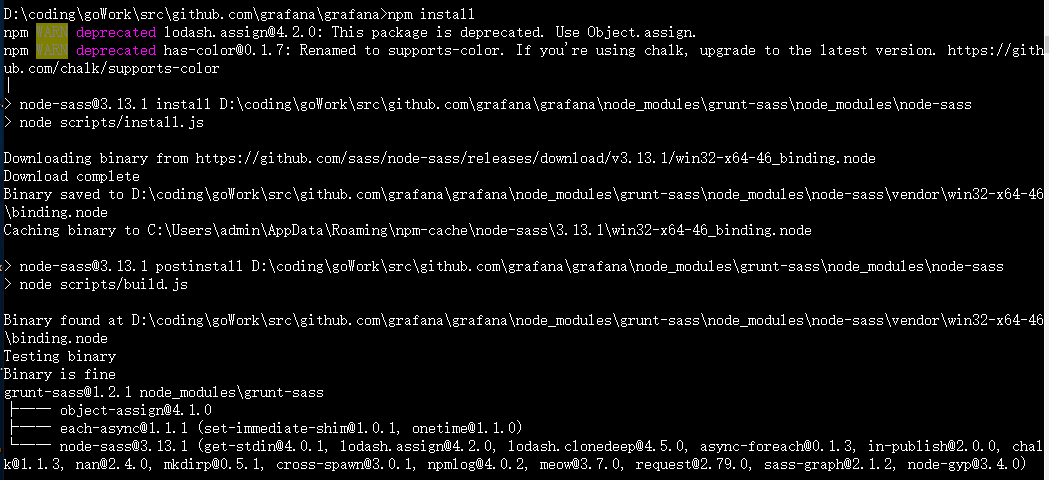
Golang中国下载：



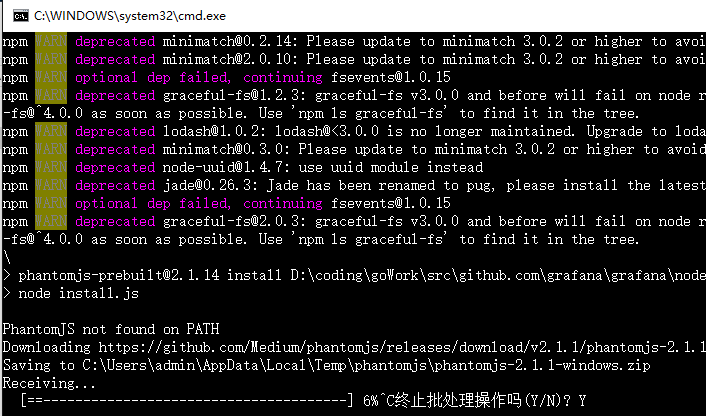


# Nodejs打包

## npm install



## npm run build



http://phantomjs.org/quick-start.html

# 其他

监测go文件改动，自动编译

go get github.com/Unknwon/bra

bra run

监测js文件改动，自动编译

sudo npm install -g grunt-cli # to do only once to install grunt command line interface

grunt watch --force

新增一个插件仍需要npm run build，若插件已新增，则只需要grunt即可，执行grunt watch

插件源码目录：%GOPATH%\src\github.com\grafana\grafana\public\app\plugins\panel

插件开发者目录：%GOPATH%\src\github.com\grafana\grafana\public\_gen\app\plugins\panel