# Battery Hostoriany的安装

<https://github.com/google/battery-historian>

安装Battery Historiany的方式有两种。

分别为Docker与源码编译安装。

对于Windows docker需要Windows 10,并且进入bios开启虚拟化。

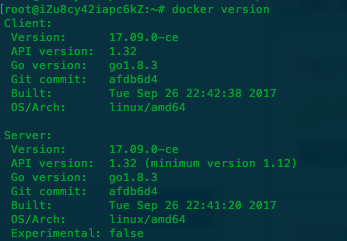
## Docker安装。

Docker 是一个开源的应用容器引擎，一般用于云计算和大数据平台，可以一键将别人发布的docket服务环境copy过来。

Docker有Community Edition (CE 社区版) and Enterprise Edition (EE 企业版)两种版本。

我们使用社区版就可以了。



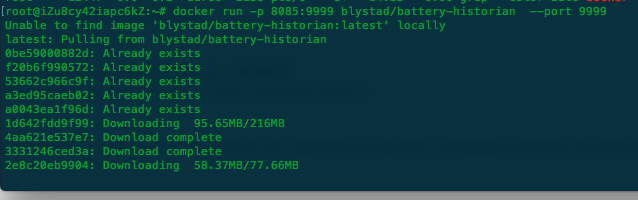


安装好Docker之后，github上让我们运行



但是这个地址貌似需要翻墙。我们可以使用镜像

docker run -p 9999:9999 blystad/battery-historian



一段漫长的等待……

然后



表示启动成功了。

## 源码编译安装。

需要存在go、git、python与java环境。

对于linux/Mac一般默认会携带python与git，如果不存在或者版本不正确也可以（brew install）。

sudo apt-get install golang

sudo apt-get install git

sudo apt-get install python2.7

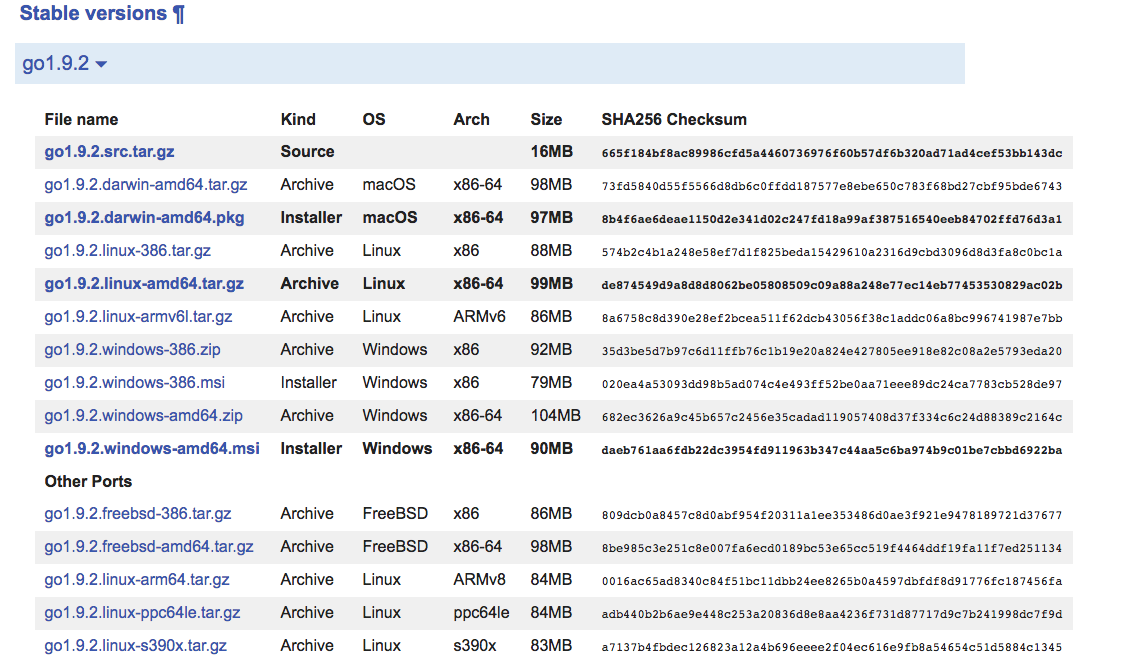
**2.1 安装go**

Go 和java一样是一个编程语言，并且是开源的。因为Battery Historiany使用了go来开发，所以我们需要安装go环境。

我们需要安装go，最低需要 1.8.1的版本

<https://golang.org/dl/>

选择自己需要安装的系统下载。第二列的Kind表示类型，有source源码(需要自己编译安装)、Installer安装器(下一步、下一步安装)与Archive编译好的(需要配置环境变量)。



安装完成后，使用go version



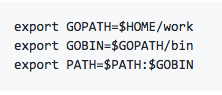
我们需要确保能够调起go工具，否则需要配置go环境变量。

Windows的在我的电脑右键属性进入配置。

Linux/Mac的则在~/.bashrc 或者 ~/.profile 或者~/.bash\_profile 都可以。

我们需要配置go的bin目录进入Path变量，这点和java一样，表示go相关工具的查找路径。另外还需要配置一个GOPATH，GOPATH就是我们工程的目录。

比如我们的工程目录在$HOME/work，则配置如下：



因为我下载的是kind 是 Instanller，所以直接配置一个GOPATH就可以了

****

****



然后保存退出。

创建这个GOPATH目录,留待使用

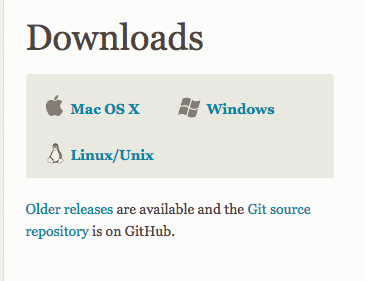


**2.2 Git工具**

<https://git-scm.com/downloads>

git工具我们应该都不陌生。和svn一样，Git是一款免费、开源的分布式版本控制系统，用于敏捷高效地处理任何或小或大的项目。

我们按照提示来进行安装就可以了



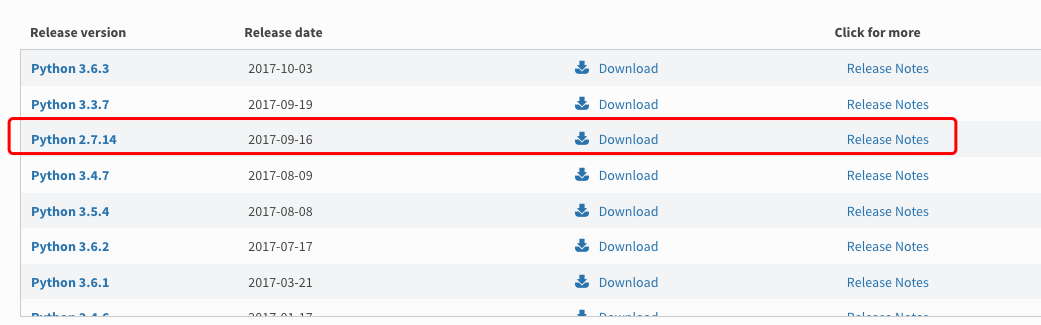
安装完成可以使用git version 检查是否按照成功。



**2.3 安装Python**

python必须使用2.7的版本，

<https://www.python.org/downloads/>



按照提示按照。同样需要配置python的目录到Path环境变量。

安装完成可以使用python –V 检查是否按照成功。



**2.4 安装java**

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

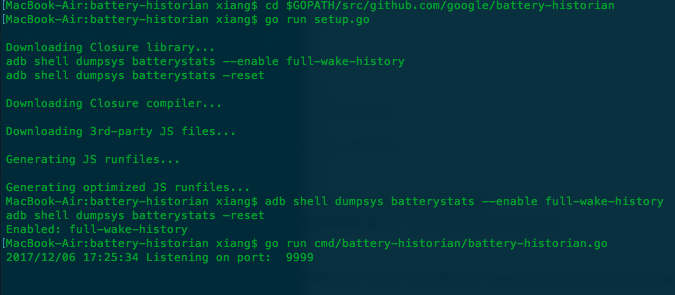
java没什么好说的了。下载安装，配置环境变量……

4个工具都安装完成之后，我们就可以下载BH源码并运行了

我们进入GOPTAH目录，下载源码与其依赖：



然后我们运行bh



cd $GOPATH/src/github.com/google/battery-historian

go run setup.go

go run cmd/battery-historian/battery-historian.go

这样就运行起来了。

以后我们需要运行，也必须每次进入GOPATH目录

cd $GOPATH/src/github.com/google/battery-historian

go run cmd/battery-historian/battery-historian.go

然后我们可以访问

[http://localhost:9999/](http://localhost:8085/)

