Server IP: 114.215.181.254

Server Pass: 4RKOZAlh48z1

SSH: ssh root@114.215.181.254

Server configuration:

OS: Ubuntu 12.04 LTS

Python, g++, Node.js:

sudo apt-get update  
sudo apt-get install -y python-software-properties python g++ make  
sudo add-apt-repository -y ppa:chris-lea/node.js  
sudo apt-get update  
sudo apt-get install nodejs

node --version : 0.10.26

Redis 2.8

sudo apt-get update

sudo apt-get install build-essential

sudo apt-get install tcl8.5

wget http://download.redis.io/releases/redis-2.8.8.tar.gz  
tar xzf redis-2.8.8.tar.gz  
cd redis-2.8.8  
make

redis configuration: <http://redis.io/topics/quickstart>

summary: copy to sbin, edit init script, init config script, start redis with init script

**Start Redis by: /etc/init.d/redis\_6379 start**

Apache2 httpd (This one took me forever, just like a year ago)

**follow**:<https://library.linode.com/web-servers/apache/installation/ubuntu-12.04-precise-pangolin>

then add Apache Proxy/Reverse Proxy:

**enable apache2 proxy module**: sudo a2enmod proxy\_http

**add following to:** /etc/apache2/apache2.conf

ProxyRequests Off

ProxyPreserveHost On

<Proxy \*>

Order allow,deny

Allow from all

</Proxy>

ProxyPass /api/ http://localhost:8015/api/

ProxyPassReverse /api/<http://localhost:8015/api/>

**then restart apache server:** sudo /etc/init.d/apache2 start || service apache2 restart

Java7:

sudo add-apt-repository ppa:webupd8team/java  
sudo apt-get update  
sudo apt-get install oracle-java7-installer

then overwrite security files to enable unlimited strength:

**find local jvm folder with command:** whereis jvm

**Copy** local\_policy.jar and US\_export\_policy.jar to the $JAVA\_HOME/jre/lib/security

Maven 3.0.4-2:

sudo apt-get install maven

settings:

Apache Maven 3.0.4

Maven home: /usr/share/maven

Java version: 1.7.0\_45, vendor: Oracle Corporation

Java home: /usr/lib/jvm/java-7-oracle/jre

Git:

sudo apt-get install git-core

Forever:

sudo npm install forever -g

Grunt

sudo npm install -g grunt-cli

Go 1.3：

GoPath: /root/go

Go installation path: /usr/local/go/

GoSEOPath：/root/goPhantom

Deployment Method:

cd /root/goPhantom

sudo ./deploy.sh

goPhantom Intro:

hotspot.txt: 专门用来存储最新的url击中记录

gen.txt: 专门用于由seoCleaner读取然后传递给phantom生成html,由hotspot移动而来，避免了读写线程冲突

deploy.sh: 用于部署seoServer和seoCleaner的脚本

phantomjs.js：用于告诉phantom生成html规则的js文件

1\*\*\*\*文件目录：对应unix epoch以来天数的文件目录，方便垃圾清理

在Apache的/etc/apache2/sites-available/default中，对应www的vhost栏目内，有以下两行：

RewriteCond %{HTTP\_USER\_AGENT} (googlebot|bingbot|yahoo|spider|baiduspider|iaskspider|Opera) [NC]

RewriteRule ^(.\*)$ http://www.ishangke.cn:8085$1 [L,P]

这里是指，将User-Agent中含有那些机器人关键词（NC表示忽略大小写）的请求，自动转发到<http://www.ishangke.cn:8085>，$1代表得到的域名后的路径， P很重要，它告诉Apache自动在内部使用proxy来进行这个新的请求，这样爬虫不需要看到302转接就能直接获得Go服务器的回复

RewriteRule ^/((?!403|404|api|a-api|p-api|test-api|test-p-api|test-a-api|build|css|js|img|countdown|scripts|design|htmlPrototype|style|targets).{1,}) #$1 [NE,L,R=301]

这一行是自动给没有#的url加上#，当然需要忽略各种静态文件的文件夹路径（这行是Harry遗作）