## 拉取镜像

docker pull jpillora/dnsmasq

## 创建文件

/opt/dnsmasq.conf

配置内容如下（ 具体配置内容可参照/etc/dnsmasq.conf ）：

domain-needed

bogus-priv

no-hosts

keep-in-foreground

no-resolv

expand-hosts

server=114.114.114.114

server=8.8.8.8

server=8.8.4.4

address=/www.opopopop.com/192.168.3.207

address=/bbs.aaa.com/192.168.3.207

server代表上游域名服务器

address代表相应的域名映射的ip

resolve-file　　   定义dnsmasq从哪里获取上游DNS服务器的地址， 默认是从/etc/resolv.conf获取。

strict-order　　   表示严格按照resolv-file文件中的顺序从上到下进行DNS解析，直到第一个解析成功为止。

listen-address     定义dnsmasq监听的地址，默认是监控本机的所有网卡上。局域网内主机若要使用dnsmasq服务时，指定本机的IP地址。

address　　　　  启用泛域名解析，即自定义解析a记录，如下配置为demon.com这个域名：

address=/demon.com/127.0.0.1 #访问demon.com时的所有域名都会被解析成127.0.0.1

bogus-nxdomain  为防止DNS污染，使用参数定义的DNS解析的服务器。注意：如果是阿里云服务器上配置dnsmasq要启用此项。

server　　　　　　指定dnsmasq程序使用哪个DNS服务器进行解析。       对于不同的网站可以使用不同的域名对应解析如下配置

server=/google.com/8.8.8.8 #表示对于google的服务，使用谷歌的DNS解析。

## 启动容器

$ docker run \

--name dnsmasq \

-d \

-p 53:53/udp \

-p 5380:8080 \

-v /opt/dnsmasq.conf:/etc/dnsmasq.conf \

--log-opt "max-size=100m" \

-e "HTTP\_USER=foo" \

-e "HTTP\_PASS=bar" \

--restart always \

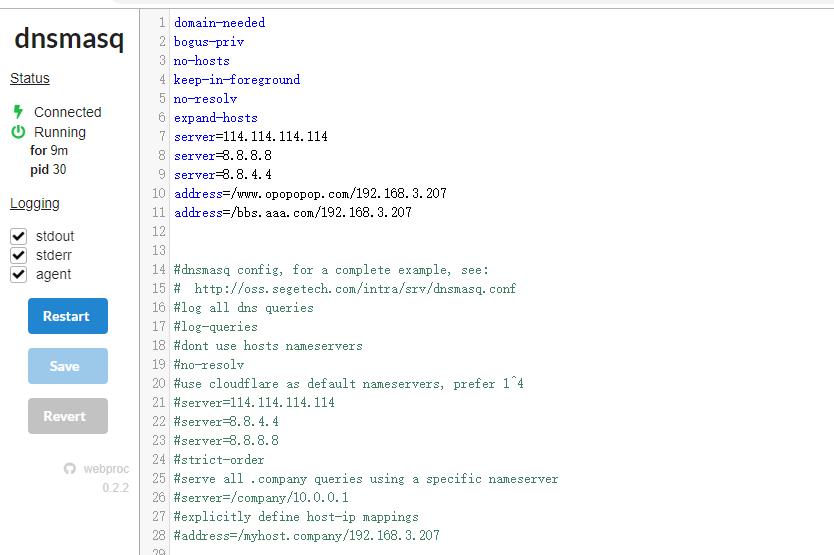
jpillora/dnsmasq

## 访问控制文件页面

http://<docker-host>:5380 页面连接地址

用户名和密码就是启动命令中设置的参数

打开后看到：



## 测试

修改需要测试的服务器resolv.conf文件

/etc/resolv.conf

在第一行插入

nameserver 192.168.0.1

将192.168.0.1替换成你的docker-dns的服务ip地址

此时该服务器引用的域名解析就是docker-dns服务的域名ip

ping bbs.aaa.com ping的地址就是192.168.3.207