nginx+lua+redis实践

概述

nginx、lua访问redis的三种方式：

1。HttpRedis模块。

指令少，功能单一，适合简单的缓存。只支持get 、select命令。

2。HttpRedis2Module模块。

功能强大，比较灵活。

3。lua-resty-redis库

OpenResty。api。适合复杂业务，节省内存。

OpenResty：基于nginx开源版本的一个扩展版本。集成了大量的精良的lua库。

OpenResty安装

安装wget

```
[root@localhost yum.repos.d]# yum install wget
```

下载资源库

```
[root@localhost yum.repos.d]# wget https://openresty.org/package/centos/openresty.repo
```

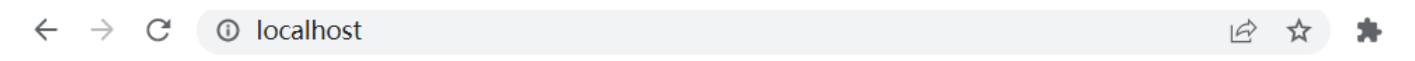得到文件： openresty.repo

安装OpenResty

```
[root@localhost yum.repos.d]# yum install openresty
```

启动OpenResty

```
[root@localhost yum.repos.d]# /usr/local/openresty/nginx/sbin/nginx -p /usr/local/openresty/nginx/
[root@localhost yum.repos.d]#
```

测试：

我们用nat，注意端口映射。

初试测试lua

修改conf

```
server {
   listen 8080;
   location / {
        default_type text/html;
        content_by_lua 'ngx.say("hello my openresty")';
   }
}
```

redis安装

安装epel:第三方的源（软件安装包）。

```
[root@localhost /]# yum install epel-release
```

安装redis

```
[root@localhost /]# yum install redis
```

启动redis

```
[root@localhost /]# /usr/bin/redis-cli
Could not connect to Redis at 127.0.0.1:6379: Connection refused
Could not connect to Redis at 127.0.0.1:6379: Connection refused
not connected>
```

```
[root@localhost /]# systemctl start redis
```

测试redis

redis服务端和redis客户端

启动redis客户端

```
[root@localhost /]# cd /usr/bin/
[root@localhost /]# which redis-cli
/usr/bin/redis-cli

[root@localhost bin]# ./redis-cli
127.0.0.1:6379> set akey avalue
OK
127.0.0.1:6379> get akey
"avalue"
127.0.0.1:6379> quit
[root@localhost bin]# pwd
/usr/bin
[root@localhost bin]#
```