# 认证服务器集群部署

## ·数据库服务器部署

### 1.1基础环境部署

可参考：<https://www.linode.com/docs/databases/mysql/how-to-install-mysql-on-centos-7/>

**安装mysql：**

rpm ivh <http://repo.mysql.com/mysql-community-release-el7-5.noarch.rpm>

yum install mysql-server

systemctl start mysqld

**进入数据库：**

执行 mysql -u root -p

回车

**创建db、用户：**

CREATE DATABASE `authserver` DEFAULT CHARACTER SET utf8 COLLATE utf8\_general\_ci;

CREATE USER 'authserver'@'%' IDENTIFIED BY 'Ct#123@Uranus';

**添加访问权限：**

GRANT ALL ON authserver.\* TO 'authserver'@'127.0.0.1' IDENTIFIED BY 'Ct#123@Uranus';

添加远程访问权限针对于认证服务器：

GRANT ALL ON authserver.\* TO 'authserver'@' 120.79.161.158 ' IDENTIFIED BY 'Ct#123@Uranus';

**初始化数据：**

use authserver;

执行[AuthServerTables.sql](https://github.com/xiliangMa/AuthServer/blob/master/packaging/dbScript/AuthServerTables.sql)文件的脚本

## 认证服务器部署

··· 2.1 安装依赖包

yum install MySQL-python.x86\_64

yum install java-1.8.0-openjdk-devel java-1.8.0-openjdk

yum install python2-jpype.x86\_64

yum install pyOpenSSL

yum install npm

npm install apidoc -g

如果报错：（npm: relocation error: npm: symbol SSL\_set\_cert\_cb, version libssl.so.10 not defined in file libssl.so.10 with link time reference）

执行 yum update openssl

### 2.2 安装python库

easy\_install pip

pip install flask==0.12.2

pip install flask-restful

pip install flask\_sqlalchemy

pip install flask\_cors

pip install flask\_httpauth

pip install qcloudsms\_py

pip install sqlalchemy

pip install pyOpenSSL

pip install gunicorn

### 2.3 安装认证服务软件：

rpm -ivh AuthServer-1.0.0-1.el7.centos.x86\_64.rpm

### 2.4 启动服务：

**配置数据源：**

vim /usr/share/AuthServer/backend/utils/SysConstant.py

修改DB\_HOST=数据库服务器ip

**启动服务：**

cd /usr/share/AuthServer

gunicorn -b 0.0.0.0:8081 AuthServer:app –reload -t 60 -D

## nginx服务器部署

* 1. 安装nginx

yum install nginx

* 1. 配置nginx集群
     1. 单服务部署将nginx.conf和server.key分别复制到以下路径：

A、/etc/nginx/nginx/key/ca.crt

B、/etc/nginx/nginx/key/server.key

* + 1. 集群部署

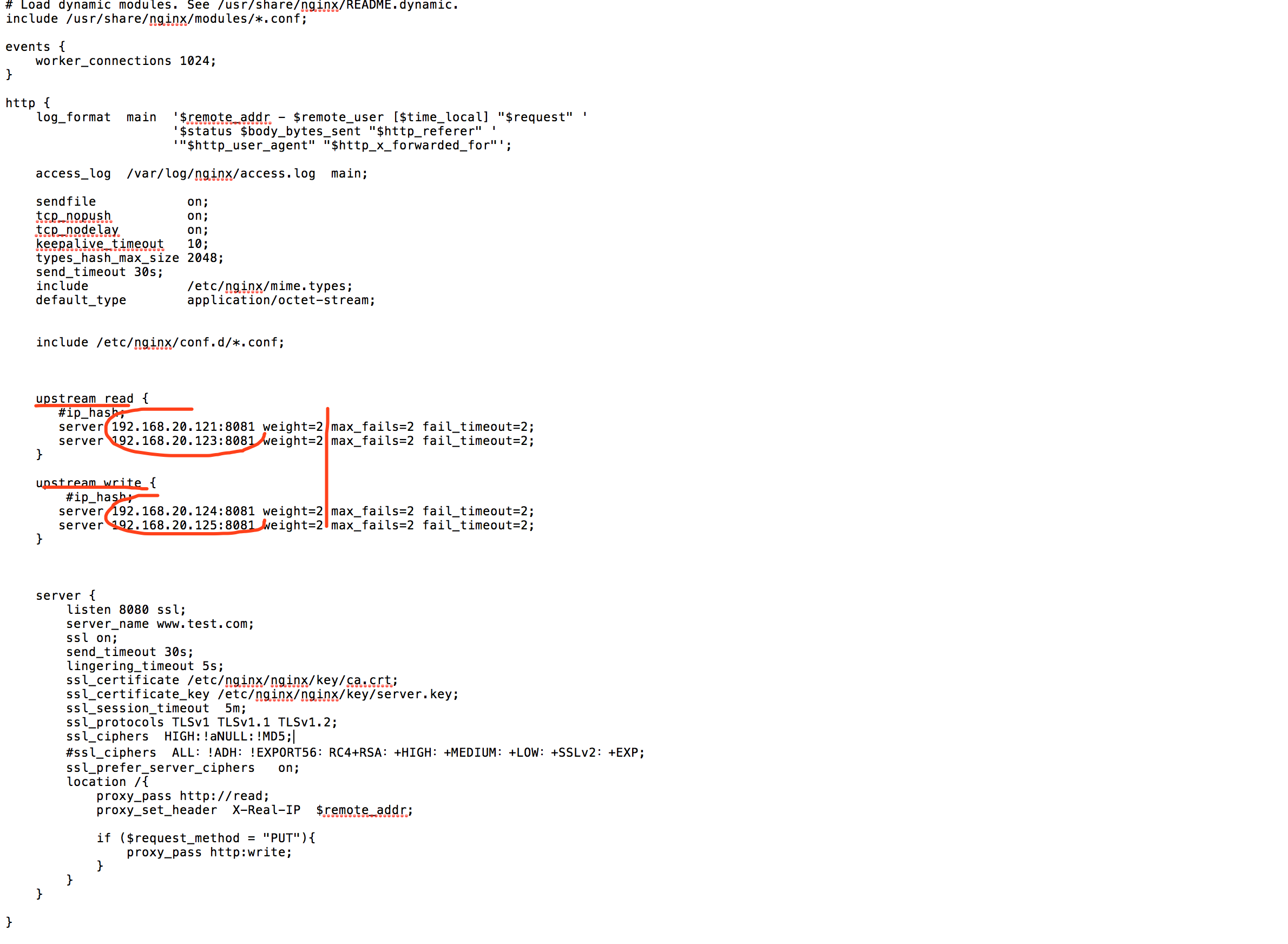
在单服务器部署基础上修改以下配置

**可以直接将nginx.conf 复制到 /etc/nginx**

**配置集群、读写分离：**

**vim /etc/nginx/nginx.conf**

**(根据自己的需要配置修改， 配置读写服务器以及配置服务器的权重)**

****

* 1. 启动nginx

service nginx restart

访问<https://nginxIp:8080/authserver>

## 对接说明

* 1. 透传服务对接

/usr/share/AuthServer/backend/extend/ppd\_client\_linux\_x64/pgTunnelSvrTool.cfg

内容如下：

(Node){

  (LclAddr){0:0:0:127.0.0.1:0:0}

  (SvrName){pgTunnelSvr0}

  (SvrAddr){211.159.180.214:7885}

  (UserName){Tst@admin}

  (UserPass){p2p@Ct#123}

  (MaxPeer){256}

  (MaxGroup){16}

  (MaxObject){512}

  (MaxMCast){512}

  (MaxHandle){256}

  (Digest){1}

}

* 1. 认证服务器系统配置

注意： 此文件主要是认证服务器系统的配置文件，

1. 数据库配置

2. 消息配置

3. 透传配置

4. 短信息配置

具体看下面的注释

/usr/share/AuthServer/backend/utils/SysConstant.py

内容如下：

# -\*- coding: utf-8 -\*-

\_\_author\_\_ = 'xiliangma'

SYSTEM\_PATH = '/usr/share/AuthServer/'

# RESULT KEY

CODE = "code"

MESSAGE = "message"

VALUE = "value"

# DATA BASE

DB\_USER\_NAME = "authserver"

DB\_PWD = "P@ssw0rd"

DB\_HOST = "127.0.0.1"

DB\_NAME = "authserver"

# requires auth

ADMIN = "admin@internal"

MD5\_PWD = "e99a18c428cb38d5f260853678922e03"

# log

LOG\_FILE\_NAME = "/var/log/authserver/authserver.log"

LOG\_PATH = "/var/log/authserver"

LOG\_UPDATE\_UNIT = "D"

LOG\_UPDATE\_TIME = 1

LOG\_BACKUP\_COUNT = 10

# PPD

PPD\_PATH = '/usr/share/AuthServer/backend/extend/ppd\_client\_linux\_x64/'

PPD\_CLASS\_PATH = PPD\_PATH + 'pgJNILib.jar:' + \

                 PPD\_PATH + 'pgAppLib.jar:' + \

                 PPD\_PATH + 'pgWebClient.jar:' + \

                 PPD\_PATH

PPD\_CONFIG = PPD\_PATH + 'pgTunnelSvrTool.cfg'

PPD\_USER\_HEAD = "\_DEV\_"

PPD\_USER\_END = "@pptun.com"

PPD\_SLEEP\_TIME = 4