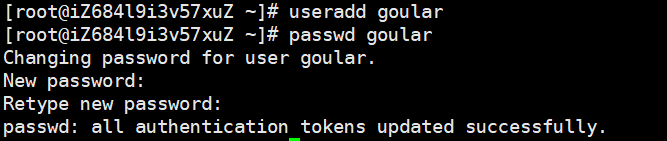
新安装的阿里云Ecs配置

1. 添加新的用户，不要使用root作为常用登录的账户



配置成功退出xShell，并以goular登录

1. 安装MariaDB

http://www.linuxidc.com/Linux/2016-03/128880.htm

1、安装MariaDB

安装命令

yum -y install mariadb mariadb-server

安装完成MariaDB，首先启动MariaDB

systemctl start mariadb

设置开机启动

systemctl enable mariadb

接下来进行MariaDB的相关简单配置

mysql\_secure\_installation

首先是设置密码，会提示先输入密码

Enter current password for root (enter for none):<–初次运行直接回车

设置密码

Set root password? [Y/n] <– 是否设置root用户密码，输入y并回车或直接回车

New password: <– 设置root用户的密码

Re-enter new password: <– 再输入一次你设置的密码

其他配置

Remove anonymous users? [Y/n] <– 是否删除匿名用户，回车

Disallow root login remotely? [Y/n] <–是否禁止root远程登录,回车,

Remove test database and access to it? [Y/n] <– 是否删除test数据库，回车

Reload privilege tables now? [Y/n] <– 是否重新加载权限表，回车

初始化MariaDB完成，接下来测试登录

mysql -uroot -ppassword

完成。

2、配置MariaDB的字符集

文件/etc/my.cnf

vi /etc/my.cnf

在[mysqld]标签下添加

init\_connect='SET collation\_connection = utf8\_unicode\_ci'

init\_connect='SET NAMES utf8'

character-set-server=utf8

collation-server=utf8\_unicode\_ci

skip-character-set-client-handshake

文件/etc/my.cnf.d/client.cnf

vi /etc/my.cnf.d/client.cnf

在[client]中添加

default-character-set=utf8

文件/etc/my.cnf.d/mysql-clients.cnf

vi /etc/my.cnf.d/mysql-clients.cnf

在[mysql]中添加

default-character-set=utf8

全部配置完成，重启mariadb

systemctl restart mariadb

之后进入MariaDB查看字符集

mysql> show variables like "%character%";show variables like "%collation%";

显示为

+--------------------------+----------------------------+

| Variable\_name | Value |

+--------------------------+----------------------------+

| character\_set\_client | utf8 |

| character\_set\_connection | utf8 |

| character\_set\_database | utf8 |

| character\_set\_filesystem | binary |

| character\_set\_results | utf8 |

| character\_set\_server | utf8 |

| character\_set\_system | utf8 |

| character\_sets\_dir | /usr/share/mysql/charsets/ |

+--------------------------+----------------------------+

8 rows in set (0.00 sec)

+----------------------+-----------------+

| Variable\_name | Value |

+----------------------+-----------------+

| collation\_connection | utf8\_unicode\_ci |

| collation\_database | utf8\_unicode\_ci |

| collation\_server | utf8\_unicode\_ci |

+----------------------+-----------------+

3 rows in set (0.00 sec)

字符集配置完成。

3、添加用户，设置权限

创建用户命令

mysql>create user username@localhost identified by 'password';

直接创建用户并授权的命令

mysql>grant all on \*.\* to username@localhost indentified by 'password';

授予外网登陆权限

mysql>grant all privileges on \*.\* to username@'%' identified by 'password';

授予权限并且可以授权

mysql>grant all privileges on \*.\* to username@'hostname' identified by 'password' with grant option;

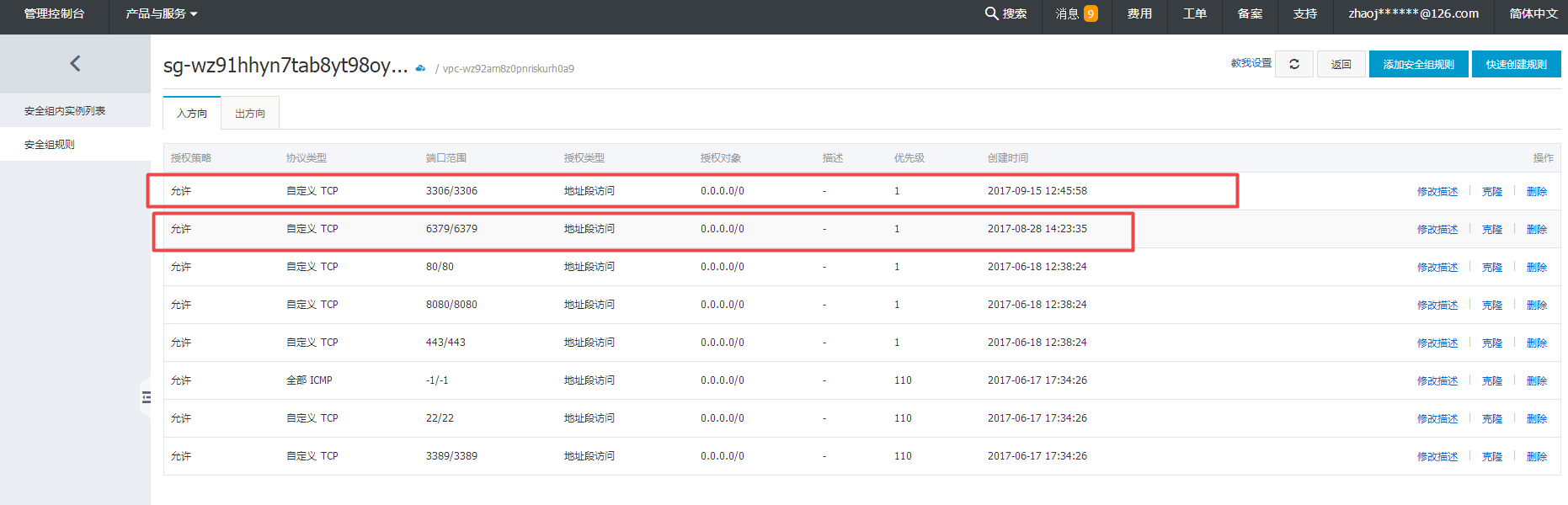
让授权配置生效（否则无法生效）

FLUSH PRIVILEGES;

简单的用户和权限配置基本就这样了。

其中只授予部分权限把 其中 all privileges或者all改为select,insert,update,delete,create,drop,index,alter,grant,references,reload,shutdown,process,file其中一部分。

阿里云会限制ECS的访问，我们需要配置:



1. 安装Nginx

一. gcc 安装

安装 nginx 需要先将官网下载的源码进行编译，编译依赖 gcc 环境，如果没有 gcc 环境，则需要安装：

yum install gcc-c++

二. PCRE pcre-devel 安装

PCRE(Perl Compatible Regular Expressions) 是一个Perl库，包括 perl 兼容的正则表达式库。nginx 的 http 模块使用 pcre 来解析正则表达式，所以需要在 linux 上安装 pcre 库，pcre-devel 是使用 pcre 开发的一个二次开发库。nginx也需要此库。命令：

yum install -y pcre pcre-devel

三. zlib 安装

zlib 库提供了很多种压缩和解压缩的方式， nginx 使用 zlib 对 http 包的内容进行 gzip ，所以需要在 Centos 上安装 zlib 库。

yum install -y zlib zlib-devel

四. OpenSSL 安装

OpenSSL 是一个强大的安全套接字层密码库，囊括主要的密码算法、常用的密钥和证书封装管理功能及 SSL 协议，并提供丰富的应用程序供测试或其它目的使用。

nginx 不仅支持 http 协议，还支持 https（即在ssl协议上传输http），所以需要在 Centos 安装 OpenSSL 库。

yum install -y openssl openssl-devel

解压

依然是直接命令：

tar -zxvf nginx-1.10.1.tar.gz

cd nginx-1.10.1

配置

其实在 nginx-1.10.1 版本中你就不需要去配置相关东西，默认就可以了。当然，如果你要自己配置目录也是可以的。

1.使用默认配置

./configure

2.自定义配置（不推荐）

./configure \

--prefix=/usr/local/nginx \

--conf-path=/usr/local/nginx/conf/nginx.conf \

--pid-path=/usr/local/nginx/conf/nginx.pid \

--lock-path=/var/lock/nginx.lock \

--error-log-path=/var/log/nginx/error.log \

--http-log-path=/var/log/nginx/access.log \

--with-http\_gzip\_static\_module \

--http-client-body-temp-path=/var/temp/nginx/client \

--http-proxy-temp-path=/var/temp/nginx/proxy \

--http-fastcgi-temp-path=/var/temp/nginx/fastcgi \

--http-uwsgi-temp-path=/var/temp/nginx/uwsgi \

--http-scgi-temp-path=/var/temp/nginx/scgi

注：将临时文件目录指定为/var/temp/nginx，需要在/var下创建temp及nginx目录

编译安装

make

make install

查找安装路径：

whereis nginx

启动、停止nginx

cd /usr/local/nginx/sbin/

./nginx

./nginx -s stop

./nginx -s quit

./nginx -s reload

./nginx -s quit:此方式停止步骤是待nginx进程处理任务完毕进行停止。

./nginx -s stop:此方式相当于先查出nginx进程id再使用kill命令强制杀掉进程。

查询nginx进程：

ps aux|grep nginx

重启 nginx

1.先停止再启动（推荐）：

对 nginx 进行重启相当于先停止再启动，即先执行停止命令再执行启动命令。如下：

./nginx -s quit

./nginx

2.重新加载配置文件：

当 ngin x的配置文件 nginx.conf 修改后，要想让配置生效需要重启 nginx，使用-s reload不用先停止 ngin x再启动 nginx 即可将配置信息在 nginx 中生效，如下：

./nginx -s reload

开机自启动

即在rc.local增加启动代码就可以了。

vi /etc/rc.local

增加一行 /usr/local/nginx/sbin/nginx

设置执行权限：

chmod 755 rc.local

到这里，nginx就安装完毕了，启动、停止、重启操作也都完成了，当然，你也可以添加为系统服务，我这里就不在演示了。

CentOS 7设置开机启动服务

建立服务文件

保存目录

设置开机自启动

其他命令

1.建立服务文件

文件路径

vim /usr/lib/systemd/system/nginx.service 1

服务文件内容

1.nginx.service

[Unit]

Description=nginx - high performance web server

After=network.target remote-fs.target nss-lookup.target

[Service]

Type=forking

ExecStart=/usr/local/nginx/sbin/nginx -c /usr/local/nginx/conf/nginx.conf

ExecReload=/usr/local/nginx/sbin/nginx -s reload

ExecStop=/usr/local/nginx/sbin/nginx -s stop

[Install]

WantedBy=multi-user.target1

2.mysql.service

[Unit]

Description=mysql

After=network.target remote-fs.target nss-lookup.target

[Service]

Type=forking

ExecStart=/usr/local/mysql/support-files/mysql.server start

#ExecReload=/usr/local/mysql/support-files/mysql.server restart

#ExecStop=/usr/local/mysql/support-files/mysql.server stop

#PrivateTmp=true

[Install]

WantedBy=multi-user.target1

3.php-fpm.service

[Unit]

Description=php

After=network.target remote-fs.target nss-lookup.target

[Service]

Type=forking

ExecStart=/usr/local/php/sbin/php-fpm

[Install]

WantedBy=multi-user.target1

4.redis.service

[Unit]

Description=Redis

After=network.target remote-fs.target nss-lookup.target

[Service]

Type=forking

ExecStart=/usr/local/bin/redis-server /etc/redis.conf

ExecStop=kill -INT `cat /tmp/redis.pid`

User=www

Group=www

[Install]

WantedBy=multi-user.target1

5.supervisord.service

[Unit]

Description=Process Monitoring and Control Daemon

After=rc-local.service

[Service]

Type=forking

ExecStart=/usr/bin/supervisord -c /etc/supervisord.conf

SysVStartPriority=99

[Install]

WantedBy=multi-user.target1

文件内容解释

[Unit]:服务的说明

Description:描述服务

After:描述服务类别

[Service]服务运行参数的设置

Type=forking是后台运行的形式

ExecStart为服务的具体运行命令

ExecReload为重启命令

ExecStop为停止命令

PrivateTmp=True表示给服务分配独立的临时空间

注意：启动、重启、停止命令全部要求使用绝对路径

[Install]服务安装的相关设置，可设置为多用户1

2.保存目录

以754的权限保存在目录：

/usr/lib/systemd/system 1

3.设置开机自启动

任意目录下执行

systemctl enable nginx.service

4.其他命令

启动nginx服务

systemctl start nginx.service

设置开机自启动

systemctl enable nginx.service

停止开机自启动

systemctl disable nginx.service

查看服务当前状态

systemctl status nginx.service

重新启动服务

systemctl restart nginx.service

查看所有已启动的服务

systemctl list-units --type=service

1. 安装php

#添加源

rpm -Uvh https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm

rpm -Uvh https://mirror.webtatic.com/yum/el7/webtatic-release.rpm

#当前关于php所有的包

yum list --enablerepo=webtatic | grep php

cd /etc/yum.repos.d

ls

yum search php71w

安装php内容

yum install php71w.x86\_64 php71w-cli.x86\_64 php71w-common.x86\_64 php71w-gd.x86\_64 php71w-ldap.x86\_64 php71w-mbstring.x86\_64 php71w-mcrypt.x86\_64 php71w-mysql.x86\_64 php71w-pdo.x86\_64

yum install php71w.x86\_64 php71w-cli.x86\_64 php71w-common.x86\_64 php71w-gd.x86\_64 php71w-ldap.x86\_64 php71w-mbstring.x86\_64 php71w-mcrypt.x86\_64 php71w-mysql.x86\_64 php71w-pdo.x86\_64php71w-cli.x86\_64php71w-devel.x86\_64 php71w-fpm.x86\_64 php71w-opcache.x86\_64 php71w-pecl-memcached.x86\_64 php71w-pecl-mongodb.x86\_64 php71w-pecl-redis.x86\_64 php71w-pecl-xdebug.x86\_64 php71w-xml.x86\_64

#添加源

rpm -Uvh https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm

rpm -Uvh https://mirror.webtatic.com/yum/el7/webtatic-release.rpm

#当前关于php所有的包

yum list --enablerepo=webtatic | grep php

cd /etc/yum.repos.d

ls

yum search php71w

安装php内容

yum install php71w.x86\_64 php71w-cli.x86\_64 php71w-common.x86\_64 php71w-gd.x86\_64 php71w-ldap.x86\_64 php71w-mbstring.x86\_64 php71w-mcrypt.x86\_64 php71w-mysql.x86\_64 php71w-pdo.x86\_64

yum install php71w.x86\_64 php71w-cli.x86\_64 php71w-common.x86\_64 php71w-gd.x86\_64 php71w-ldap.x86\_64 php71w-mbstring.x86\_64 php71w-mcrypt.x86\_64 php71w-mysql.x86\_64 php71w-pdo.x86\_64php71w-cli.x86\_64php71w-devel.x86\_64 php71w-fpm.x86\_64 php71w-opcache.x86\_64 php71w-pecl-memcached.x86\_64 php71w-pecl-mongodb.x86\_64 php71w-pecl-redis.x86\_64 php71w-pecl-xdebug.x86\_64 php71w-xml.x86\_64

systemctl enable \*.service #开机运行服务

systemctl disable \*.service #取消开机运行

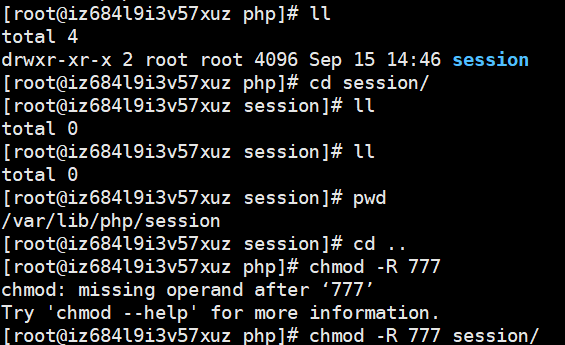
systemctl start \*.service #启动服务

systemctl stop \*.service #停止服务

systemctl restart \*.service #重启服务

### 安装php7.1的小bug





1. 安装redis

http://www.cnblogs.com/sandea/p/5782192.html

1. 安装beanstalk

切换到 root用户下：

yum install libevent libevent-devel

所有版本地址： https://github.com/kr/beanstalkd/downloads

下载地址：

wget http://cloud.github.com/downloads/kr/beanstalkd/beanstalkd-1.4.6.tar.gz （或附件的beanstalkd-1.4.6.tar.gz）

解压：

tar xzf beanstalkd-1.4.6.tar.gz

cd beanstalkd-1.4.6

./configure

make

make install

默认安装路径 ：/usr/local/bin/

查看版本：

/usr/local/bin/beanstalkd -v

1.4.6

添加用户组：

groupadd beanstalkd

添加用户：

adduser -M -g beanstalkd -s /sbin/nologin beanstalkd

添加启动脚本，放入 /etc/init.d/ 目录下