Linux下配置Shadowsocks服务器的步骤详解

Shadowsocks是一款代理的软件了我们可以在linux机器配置好Shadowsocks之后再进入看国外网站了，下面一起来看Shadowsocks配置教程。

说明：

Shadowsocks是一个轻量级的socks5代理软件，

而hadowsocks-libev是一个基于shadowsocks 协议的socks5代理软件，

相比原版，hadowsocks-libev程序体积小、高并发、资源占用更少、跨平台、完全兼容shadowsocks协议。

hadowsocks-libev包括三个模块：

ss-server：服务器端，部署在远程服务器，提供shadowsocks服务。

ss-local：客户端，提供本地socks5协议代理。

ss-redir：客户端，提供本地透明代理。

实现目的：

用一台国外VPS服务器，安装部署hadowsocks-libev代理软件，让国内用户通过这台VPS服务器能够访问国外网站。

具体操作：

注意：服务器先安装git软件，然后通过git软件在线下载hadowsocks-libev软件，最后，再编译安装hadowsocks-libev。

VPS服务器操作系统：CentOS

VPS服务器ip：192.168.1.161

一、关闭SELINUX

vi /etc/selinux/config

#SELINUX=enforcing #注释掉

#SELINUXTYPE=targeted #注释掉

SELINUX=disabled #增加

:wq! #保存退出

setenforce 0 #使配置立即生效

二、开启防火墙端口（TCP 8388端口为hadowsocks-libev服务端默认端口）

vi /etc/sysconfig/iptables #编辑防火墙配置文件

# Firewall configuration written by system-config-securitylevel

# Manual customization of this file is not recommended.

\*filter

:INPUT ACCEPT [0:0]

:FORWARD ACCEPT [0:0]

:OUTPUT ACCEPT [0:0]

:RH-Firewall-1-INPUT - [0:0]

-A INPUT -j RH-Firewall-1-INPUT

-A FORWARD -j RH-Firewall-1-INPUT

-A RH-Firewall-1-INPUT -i lo -j ACCEPT

-A RH-Firewall-1-INPUT -p icmp --icmp-type any -j ACCEPT

-A RH-Firewall-1-INPUT -p 50 -j ACCEPT

-A RH-Firewall-1-INPUT -p 51 -j ACCEPT

-A RH-Firewall-1-INPUT -p udp --dport 5353 -d 224.0.0.251 -j ACCEPT

-A RH-Firewall-1-INPUT -p udp -m udp --dport 631 -j ACCEPT

-A RH-Firewall-1-INPUT -p tcp -m tcp --dport 631 -j ACCEPT

-A RH-Firewall-1-INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT

-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 8388 -j ACCEPT

-A RH-Firewall-1-INPUT -j REJECT --reject-with icmp-host-prohibited

COMMIT

:wq! #保存退出

service iptables restart #最后重启防火墙使配置生效

三、安装编辑工具包

1、CentOS 5.x下编译安装git需要的包（默认CentOS 5.x yum源中没有git）

yum install wget curl curl-devel zlib-devel openssl-devel perl perl-devel cpio expat-devel gettext-devel

2、安装shadowsocks-libev.git需要的包

yum install autoconf libtool openssl-devel gcc swig python-devel

四、安装git

1、CentOS 5.x下安装

cd /usr/local/src

wget https://git-core.googlecode.com/files/git-1.9.0.tar.gz #此地址需要翻墙，可以先下载好软件之后，再上传到服务器

tar xzvf git-1.9.0.tar.gz #解压

cd git-1.9.0 #进入安装目录

autoconf

./configure #配置

make #编译

make install #安装

git --version #查看版本

2、CentOS 6.x下安装

yum install git #CentOS6中yum源中已经有git的版本，直接yum安装

五、安装shadowsocks-libev

cd /usr/local/src

git clone https://github.com/madeye/shadowsocks-libev.git #使用git下载

cd shadowsocks-libev #进入安装目录

./configure #配置

make #编译

make install #安装

cd /usr/local/bin/ #安装好之后，会在此目录生成以下文件

ss-server

ss-local

ss-redir

六、配置shadowsocks-libev服务端

系统运维 www.osyunwei.com 温馨提醒：qihang01原创内容©版权所有,转载请注明出处及原文链接

1、创建配置文件

mkdir /etc/shadowsocks-libev #创建配置文件存放目录

vi /etc/shadowsocks-libev/config.json #编辑，添加以下内容

{

"server":"192.168.1.161", #服务端监听的IP地址

"server\_port":8388, #服务端端口

"local\_address":"127.0.0.1", #本地监听的IP地址

"local\_port":1080, #本地端端口

"password":"123456", #用来加密的密码

"timeout":60, #超时时间（秒）

"method":"aes-256-cfb", #加密方法，推荐用 “aes-256-cfb”

}

:wq! #保存退出

2、运行shadowsocks-libev服务端

nohup /usr/local/bin/ss-server -u -c /etc/shadowsocks-libev/config.json -f /var/run/shadowsocks-server/pid &

3、关闭shadowsocks-libev服务端

ps -ef|grep ss-server #查看进程ID

kill -9 进程ID #结束shadowsocks-libev服务端

4、设置shadowsocks-libev服务端开机启动

vi /etc/init.d/ss-server #编辑，添加以下代码

#!/bin/bash

# Run level information:

# chkconfig: 2345 99 99

# Description: lightweight secured socks5 proxy

# processname: ss-server

# Paths and variables and system checks.

# Source function library

. /etc/rc.d/init.d/functions

# Check that networking is up.

#

[ ${NETWORKING} ="yes" ] || exit 0

# Daemon

NAME=shadowsocks-server

DAEMON=/usr/local/bin/ss-server

# Path to the configuration file.

#

CONF=/etc/shadowsocks-libev/config.json

#USER="nobody"

#GROUP="nobody"

# Take care of pidfile permissions

mkdir /var/run/$NAME 2>/dev/null || true

#chown "$USER:$GROUP" /var/run/$NAME

# Check the configuration file exists.

#

if [ ! -f $CONF ] ; then

echo "The configuration file cannot be found!"

exit 0

fi

# Path to the lock file.

#

LOCK\_FILE=/var/lock/subsys/shadowsocks

# Path to the pid file.

#

PID=/var/run/$NAME/pid

#====================================================================

#====================================================================

# Run controls:

RETVAL=0

# Start shadowsocks as daemon.

#

start() {

if [ -f $LOCK\_FILE ]; then

echo "$NAME is already running!"

exit 0

else

echo -n $"Starting ${NAME}: "

#daemon --check $DAEMON --user $USER "$DAEMON -f $PID -c $CONF > /dev/null"

daemon $DAEMON -u -c $CONF -f $PID

fi

RETVAL=$?

[ $RETVAL -eq 0 ] && success

echo

[ $RETVAL -eq 0 ] && touch $LOCK\_FILE

return $RETVAL

}

# Stop shadowsocks.

#

stop() {

echo -n $"Shutting down ${NAME}: "

killproc -p ${PID}

RETVAL=$?

[ $RETVAL -eq 0 ]

rm -f $LOCK\_FILE

rm -f ${PID}

echo

return $RETVAL

}

# See how we were called.

case "$1" in

start)

start

;;

stop)

stop

;;

restart)

stop

start

;;

condrestart)

if [ -f $LOCK\_FILE ]; then

stop

start

RETVAL=$?

fi

;;

status)

status $DAEMON

RETVAL=$?

;;

\*)

echo $"Usage: $0 {start|stop|restart|condrestart|status}"

RETVAL=1

esac

exit $RETVAL

:wq! #保存退出

chmod +x /etc/init.d/ss-server #添加脚本执行权限

chkconfig --add ss-server #添加到开机启动

chkconfig ss-server on #设置开机启动

相关命令：

启动：/etc/init.d/ss-server start

停止：/etc/init.d/ss-server stop

重启：/etc/init.d/ss-server restart

查看状态：/etc/init.d/ss-server status

七、配置shadowsocks-libev客户端

1、Windows下安装Shadowsocks客户端

下载地址：

http://sourceforge.net/projects/shadowsocksgui/files/dist/Shadowsocks-win-2.3.1.zip

下载好之后，打开运行，如下图所示：

服务器IP：192.168.1.161

服务器端口：8388

密码：123456

加密：aes-256-cfb

备注：

代理端口：1080

点击确定

注意：这一步只是用Shadowsocks客户端连接上了服务器，还需要在浏览器里面设置代理之后，才能上网。

2、打开浏览器设置代理（推荐使用Google和火狐浏览器）

下面以火狐浏览器为例：

打开火狐浏览器

选项-高级-网络-设置-连接设置-手动配置代理

SOCKS主机：192.168.1.161

端口：1080

选择SOCKS v5

设置好之后，点确定。

现在，已经可以正常打开google等国外网站了！

八、把Shadowsocks客户端配置在Shadowsocks服务器上面，让用户不需要安装Shadowsocks客户端，直接设置浏览器代理就能访问国外网站。

以下在Shadowsocks服务器上操作

此时，需要在服务器上开放Shadowsocks客户端端口TCP 1080（开放端口方法同上面一样）

1、设置配置文件

vi /etc/shadowsocks-libev/shadowsocks.json #编辑，添加以下代码

{

"server":"192.168.1.161",

"server\_port":8388,

"local\_address":"127.0.0.1",

"local\_port":1080,

"password":"123456",

"timeout":60,

"method":"aes-256-cfb",

}

:wq! #保存退出

2、启动Shadowsocks客户端

nohup /usr/local/bin/ss-local -u -c /etc/shadowsocks-libev/shadowsocks.json -f /var/run/shadowsocks-local/pid -b 0.0.0.0 &

3、关闭shadowsocks-libev客户端

ps -ef|grep ss-local #查看进程ID

kill -9 进程ID #结束shadowsocks-libev客户端

4、设置shadowsocks-libev客户端开机启动

vi /etc/init.d/ss-local #编辑，添加以下文件

#!/bin/bash

# Run level information:

# chkconfig: 2345 88 88

# Description: lightweight secured socks5 proxy

# processname: ss-local

# Paths and variables and system checks.

# Source function library

. /etc/rc.d/init.d/functions

# Check that networking is up.

#

[ ${NETWORKING} ="yes" ] || exit 0

# Daemon

NAME=shadowsocks-local

DAEMON=/usr/local/bin/ss-local

# Path to the configuration file.

#

CONF=/etc/shadowsocks-libev/shadowsocks.json

#USER="nobody"

#GROUP="nobody"

# Take care of pidfile permissions

mkdir /var/run/$NAME 2>/dev/null || true

#chown "$USER:$GROUP" /var/run/$NAME

# Check the configuration file exists.

#

if [ ! -f $CONF ] ; then

echo "The configuration file cannot be found!"

exit 0

fi

# Path to the lock file.

#

LOCK\_FILE=/var/lock/subsys/shadowsocks

# Path to the pid file.

#

PID=/var/run/$NAME/pid

#====================================================================

#====================================================================

# Run controls:

RETVAL=0

# Start shadowsocks as daemon.

#

start() {

if [ -f $LOCK\_FILE ]; then

echo "$NAME is already running!"

exit 0

else

echo -n $"Starting ${NAME}: "

#daemon --check $DAEMON --user $USER "$DAEMON -f $PID -c $CONF > /dev/null"

daemon $DAEMON -u -c $CONF -f $PID -b 0.0.0.0

fi

RETVAL=$?

[ $RETVAL -eq 0 ] && success

echo

[ $RETVAL -eq 0 ] && touch $LOCK\_FILE

return $RETVAL

}

# Stop shadowsocks.

#

stop() {

echo -n $"Shutting down ${NAME}: "

killproc -p ${PID}

RETVAL=$?

[ $RETVAL -eq 0 ]

rm -f $LOCK\_FILE

rm -f ${PID}

echo

return $RETVAL

}

# See how we were called.

case "$1" in

start)

start

;;

stop)

stop

;;

restart)

stop

start

;;

condrestart)

if [ -f $LOCK\_FILE ]; then

stop

start

RETVAL=$?

fi

;;

status)

status $DAEMON

RETVAL=$?

;;

\*)

echo $"Usage: $0 {start|stop|restart|condrestart|status}"

RETVAL=1

esac

exit $RETVAL

:wq! #保存退出

chmod +x /etc/init.d/ss-local #添加脚本执行权限

chkconfig --add ss-local #添加开机启动

chkconfig ss-local on #设置开机启动

启动：/etc/init.d/ss-local start

停止：/etc/init.d/ss-local stop

重启：/etc/init.d/ss-local restart

查看状态：/etc/init.d/ss-local status

设置完成之后，用户直接在浏览器里面设置代理（步骤同上面一样）即可访问国外网站。

扩展阅读：

1、可以专门用一台国内服务器当做Shadowsocks客户端服务器，用来连接国外的Shadowsocks服务器，然后，用户浏览器里面直接设置国内服务器为代理IP。

2、用户浏览器端还能设置自动代理配置，在google浏览器里面先把配置好的文件导出为.pac，然后把这个.pac文件放到网站上，最后给用户提供一个.pac的url地址，用户在浏览器中只需要把这个url地址填写到自动代理配置中即可。

至此，Linux下配置Shadowsocks代理服务器浏览国外网站教程完成。