# FirewallSetting linux服务器部署

## 安装

#### 1.1安装.net

请按照微软.net core安装.net 5.0

<https://docs.microsoft.com/zh-cn/dotnet/core/install/linux>

#### 1.2安装supervisord(CentOS)

yum install -y epel-release

yum install -y supervisor

systemctl enable supervisord.service

systemctl start supervisord.service

systemctl restart supervisord.service

**===查看状态===**

systemctl status supervisord.service

**===打开Web界面===**

/etc/supervisord.conf

头部去掉注释

[inet\_http\_server] ; inet (TCP) server disabled by default

port=0.0.0.0:9001 ; (ip\_address:port specifier, \*:port for all iface)

username=user ; (default is no username (open server))

password=123 ; (default is no password (open server))

**===其他命令===**

supervisorctl fg firewallsetting //进入管理firewallsetting程序

supervisorctl start firewallsetting //打开firewallsetting程序

supervisorctl stop firewallsetting //关闭firewallsetting程序

supervisorctl restart firewallsetting //重新firewallsetting程序

#### 1.3部署到supervisord

先把**FirewallSetting**所有文件拷贝到**/opt/FirewallSetting/**

然后到 **/etc/supervisord.d/** 新建一个 **firewallsetting.ini**

**===内容为：===**

[program:firewallsetting]

command=dotnet FirewallSettingServerCore.dll ; 输入执行命令，这里表示 dotnet Deploy.Linux.dll

directory=/opt/FirewallSetting/ ; 应用程序根目录

autostart=true ; 是否自动启动，当 supervisor 加载该配置文件的时候立即启动它

autorestart=true ; 是否自动重启，当执行 dotnet FirewallSettingServerCore.dll 启动失败时，会重复的自动重启

logfile\_maxbytes=50MB ; 该配置文件输出单个日志文件的大小

logfile\_backups=10 ; 日志备份个数

loglevel=info ; 记录日志级别

stderr\_logfile=/data/logs/FirewallSetting/FirewallSetting.err.log ; 指定标准错误输出日志文件

stdout\_logfile=/data/logs/FirewallSetting/FirewallSetting.out.log ; 指定标准输出日志文件

environment=ASPNETCORE\_ENVIRONMENT=Production ; 可配置环境变量，该环境变量将通过执行 dotnet Deploy.Linux.dll 命令的时候传入到 .NET Core 应用程序中

user=root ;启动服务的用户

stopsignal=INT

redirect\_stderr=true

## 配置

#### firewallRule.xml配置

firewallRule是本项目的防火墙的规则配置，每条一个防火墙规则如下：

<?xml version="1.0" encoding="utf-8" ?>

<root>

<rule name="supervisor查看" port="9001" protocol="tcp,udp"/>

<rule name="MySQL" port="33169" protocol="tcp,udp"/>

</root>

name未规则名(随意非空)

port:为端口(1-65535)

protocol:协议用逗号(,)隔开(tcp,udp)

#### 2.2主配置：FirewallSettingServerCore.dll.config

<?xml version="1.0" encoding="utf-8" ?>

<configuration>

<appSettings>

<add key="Server.Name" value="CentOS测试"/>

<add key="Server.URL" value="http://192.168.31.28:8789"/>

<add key="Server.Listen" value="http://+:8789/"/>

<add key="Server.AllowIP" value="192.168.31.25"/>

<add key="App.IPSet" value=" "/>

<!--默认添加的白名单IP,用,隔开-->

<add key="App.CacheType" value="redis"/>

<add key="App.Cache" value="server=192.168.1.10:6379;throw=0;database=3"/>

<add key="SSH.UserName" value="root"/>

<add key="SSH.UserPassword" value="111111"/>

<add key="SSH.PrivateKey" value="/opt/FirewallSetting/serverKey.pem"/>

<add key="SSH.Host" value="127.0.0.1"/>

<add key="SSH.Port" value="22"/>

<add key="App.UseSudo" value="0"/>

<add key=" App.NFTableName" value="0"/>

<add key=" App.NFChain" value="0"/>

</appSettings>

</configuration>

Server.Name：服务器名，会显示在客户端的条目

Server.URL：客户机访问这个服务器时候使用的地址

Server.Listen：监听地址

Server.AllowIP：默认加到白名单的地址，用逗号(,)隔开

Server.FirewallType：跳过检测强制使用此防火墙方案（可选参数：iptables、firewalld）

App.IPSet:生成的IP集的名字，不填则默认生成

App.CacheType:本项目使用的缓存(redis,web)，不填则使用内存

App.Cache：App.CacheType的连接字符串，App.CacheType为空或system时候此处留空

SSH.UserName:登录到本服务器的SSH名

SSH. UserPassword:登录密码，如果是**PrivateKey模式登录，则此处删除**

SSH.PrivateKey:PrivateKey路径，**如果是用户名密码登录则此处删除**

SSH.Host:服务器内网地址，本机则为localhost或127.0.0.1

SSH.Port:SSH服务的端口

App.UseSudo 使用sudo执行指令，ubuntu需要

App.NFTableName用来创建规则的table（仅限于NFTable，不指定的话，名字默认为firewalld）

App.NFChain 用来创建规则的Chain（仅限于NFTable，不指定的话，名字默认为filter\_IN\_public\_allow）

## 使用

在SSH界面键入 **supervisorctl fg firewallsetting** 进入管理

按下回车看见

命令:

可以输入命令

命令列表：

exit：关闭程序，supervisorct接管下不可用

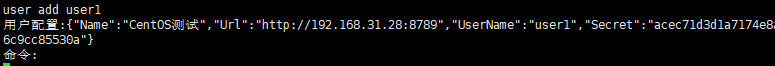
refreash：手动刷新防火墙规则

clear:清屏，supervisorct接管下不可用

user show 用户名 ：显示用户的配置json以复制到客户端



user add 用户名：新增用户，并且显示其配置json



user renew 用户名：立刻随机更换此用户的安全码并显示

