jumpserver 跳板机

Jumpserver 是全球首款完全开源的堡垒机，使用 GNU GPL v2.0 开源协议，是符合 4A 的专业运维审计系统。

Jumpserver 使用 Python3 / Django 进行开发，遵循 Web 2.0 规范，配备了业界领先的 Web Terminal 解决方案，交互界面美观、用户体验好。

作用：

1、身份鉴别

2、访问控制

3、安全审计

jumpserver的组件：

1. Jumpserver为管理后台，管理员可以通过Web页面进行资产管理、用户管理、资产授权等操作
2. Coco 为 SSH Server 和 Web Terminal Server 。用户可以通过使用自己的账户登录 SSH 或者 Web Terminal 直接访问被授权的资产。不需要知道服务器的账户密码
3. Luna 为 Web Terminal Server 前端页面，用户使用 Web Terminal 方式登录所需要的组件
4. Guacamole 为 Windows 组件，用户可以通过 Web Terminal 来连接 Windows 资产 （暂时只能通过 Web Terminal 来访问）

**部署jumserver**

**1、配置CentOS/epel 软件仓库**

[root@jumpserver ~]# yum install -y epel-release

**2、将系统语言配置为中文**

[root@jumpserver ~]# vim /etc/profile

export LANG=zh\_CN.UTF-8

[root@jumpserver ~]# source /etc/profile

[root@jumpserver ~]# echo $LANG

zh\_CN.UTF-8

**3、安装python3解释器**

[root@jumpserver ~]# yum -y install wget sqlite-devel xz gcc automake zlib-devel openssl-devel git

1) 编译安装python 3

[root@jumpserver ~]# tar xf Python-3.6.4.tar.xz

[root@jumpserver ~]# cd Python-3.6.4/

[root@jumpserver Python-3.6.4]# ./configure && make && make install

[root@jumpserver ~]# python3

Python 3.6.4 (default, Sep 17 2018, 10:13:58)

[GCC 4.8.5 20150623 (Red Hat 4.8.5-28)] on linux

Type "help", "copyright", "credits" or "license" for more information.

>>> exit()

2) 创建python3的虚拟环境

[root@jumpserver ~]# cd /opt/

[root@jumpserver opt]# python3 -m venv py3

3) 进入py3的虚拟环境

[root@jumpserver ~]# source /opt/py3/bin/activate

(py3) [root@jumpserver ~]#

**4、安装jumpserver**

(py3) [root@jumpserver ~]# cd /opt/

(py3) [root@jumpserver opt]# unzip jumpserver-master.zip

1) 安装rpm的依赖包

(py3) [root@jumpserver requirements]# cd /opt/jumpserver/requirements

(py3) [root@jumpserver requirements]# yum install -y $(cat rpm\_requirements.txt)

2) 安装依赖的python模块

(py3) [root@jumpserver requirements]# pip install -r requirements.txt -i https://pypi.python.org/simple

以上依赖中可能会遇到python-gssapi模块装不上，需要手动通过如下命令安装

(py3) [root@jumpserver ~]# pip install python-gssapi==0.6.4

3) 安装redis数据库，作为jumpserver的缓存服务器

(py3) [root@jumpserver requirements]# yum install -y redis

(py3) [root@jumpserver requirements]# systemctl start redis

(py3) [root@jumpserver requirements]# systemctl enable redis

(py3) [root@jumpserver requirements]# netstat -antp | grep redis

tcp 0 0 127.0.0.1:6379 0.0.0.0:\* LISTEN 19917/redis-server

4) 安装MySQL数据库，作为jumpserver的后台存储服务器

(py3) [root@jumpserver requirements]# yum install -y mariadb-server mariadb-devel mariadb

(py3) [root@jumpserver requirements]# systemctl start mariadb

(py3) [root@jumpserver requirements]# systemctl enable mariadb

MariaDB [(none)]> create database jumpserver charset utf8;

MariaDB [(none)]> grant all on jumpserver.\* to 'jumpserver'@'127.0.0.1' identified by 'redhat';

MariaDB [(none)]> flush privileges;

5) 编辑jumpserver的配置文件

(py3) [root@jumpserver ~]# cd /opt/jumpserver-master/

(py3) [root@jumpserver jumpserver-master]# cp config\_example.py config.py

18 SECRET\_KEY = '2vym+ky!997d5kkcc64mnz06y1mmui3lut#(^wd=%s\_qj$1%x' //指定加/解密数据的随机数

21 ALLOWED\_HOSTS = ['\*'] //允许所有主机访问web页面

24 DEBUG = True //启用debug调试

27 LOG\_LEVEL = 'WARNING'

注释自带的sqlite数据库配置,启用mysql的配置

38 DB\_ENGINE = 'mysql'

39 DB\_HOST = '127.0.0.1'

40 DB\_PORT = 3306

41 DB\_USER = 'jumpserver'

42 DB\_PASSWORD = 'redhat'

43 DB\_NAME = 'jumpserver'

配置redis数据库信息

51 REDIS\_HOST = '127.0.0.1'

52 REDIS\_PORT = 6379

53 REDIS\_PASSWORD = ''

54 REDIS\_DB\_CELERY = 3

55 REDIS\_DB\_CACHE = 4

6) 在mysql生成jumpserver需要的表

(py3) [root@jumpserver jumpserver-master]# cd /opt/jumpserver-master/utils/

(py3) [root@jumpserver utils]# bash make\_migrations.sh

7) 启动jumpserver

(py3) [root@jumpserver utils]# cd /opt/jumpserver-master/

(py3) [root@jumpserver jumpserver-master]# python3 run\_server.py &

(py3) [root@jumpserver jumpserver-master]# netstat -antp | grep :8080

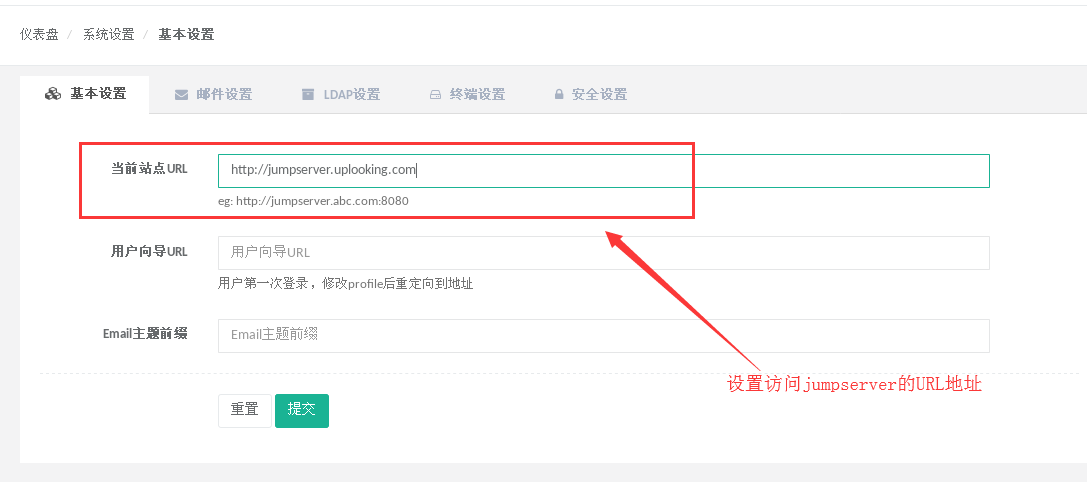
tcp 0 0 0.0.0.0:8080 0.0.0.0:\* LISTEN 20489/python3

在浏览器访问8080端口,打开jumpserver提供的web页面

http://192.168.122.101:8080

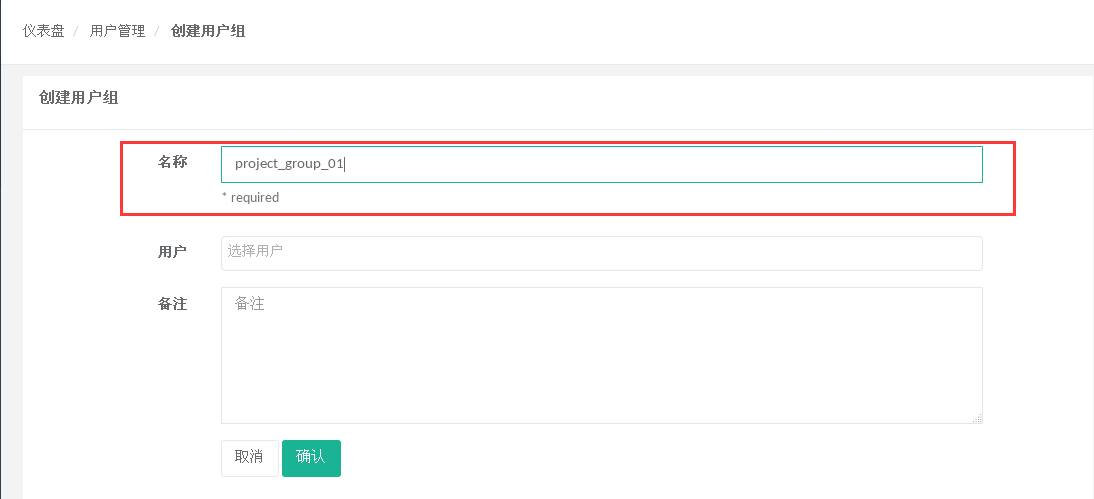
使用默认的用户名admin, 密码admin登录

**基本设置**



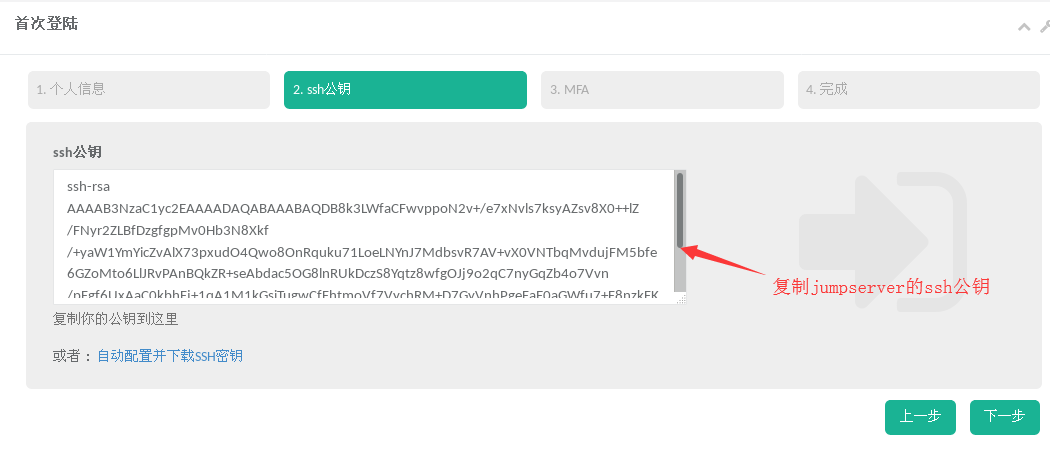
**用户管理**

创建登录jumpserver的用户名/用户组





新建的用户首次登录web界面,需要完善用户信息

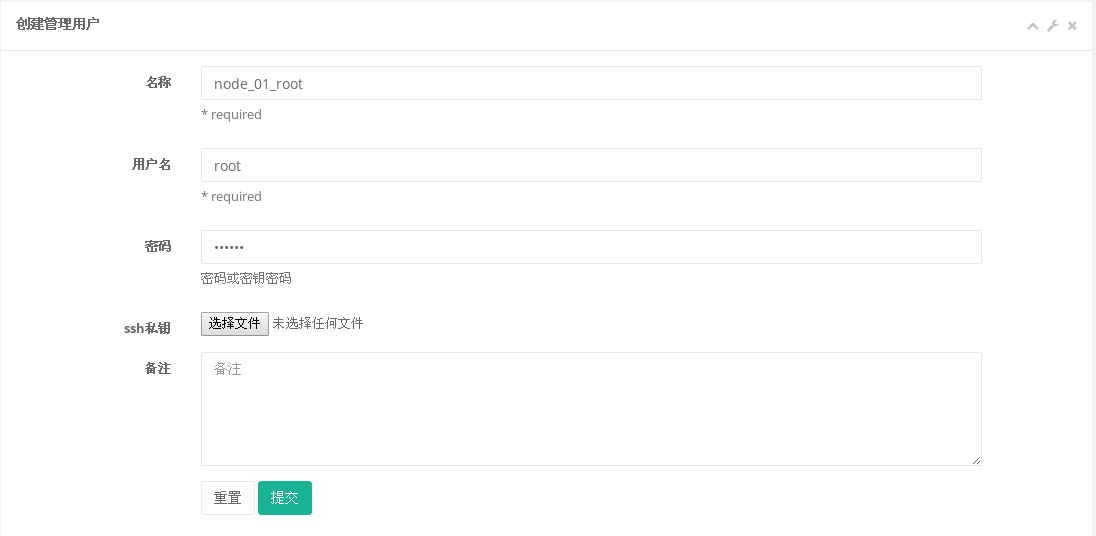


**资产管理**

创建管理用户

jumpserver连接服务器的账号

可以是root或者NOPASSWD:ALL的普通用户

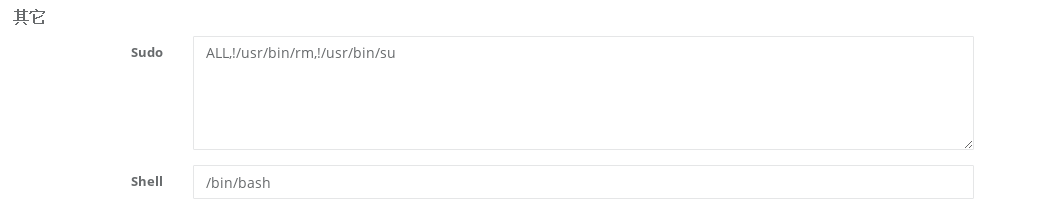


创建资产

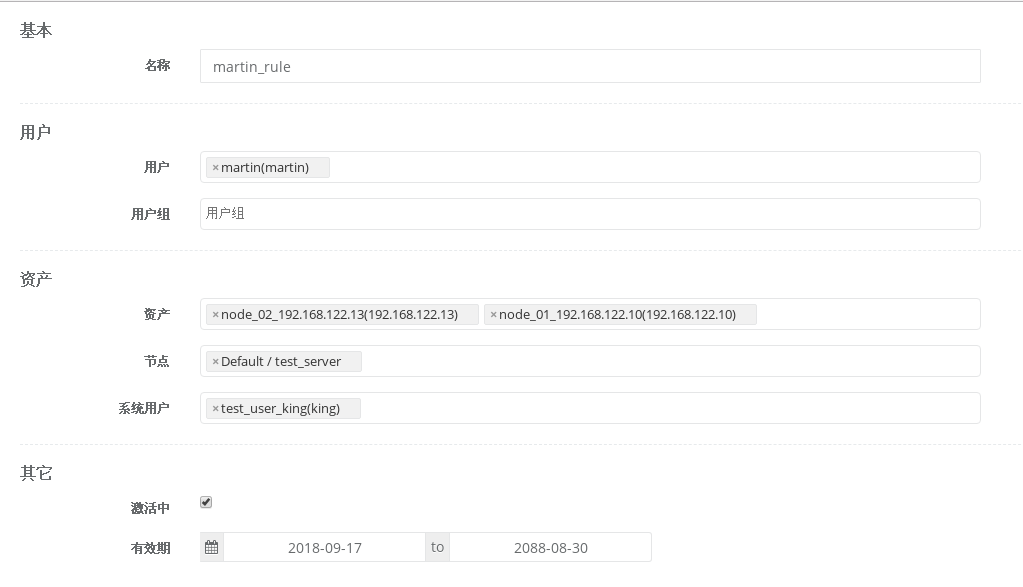
**系统用户**

跳板机代替用户远程连接服务器时需要用的用户





**资产授权**



**在jumpserver上安装coco组件提供ssh服务**

**1、下载coco组件**

(py3) [root@jumpserver opt]# cd /opt/

(py3) [root@jumpserver opt]# git clone https://github.com/jumpserver/coco.git

2、安装coco组件依赖

(py3) [root@jumpserver coco]# cd /opt/coco/requirements/

(py3) [root@jumpserver requirements]# yum install -y $(cat rpm\_requirements.txt)

(py3) [root@jumpserver requirements]# pip install -r requirements.txt -i https://pypi.python.org/simple

3、编辑coco的配置文件

(py3) [root@jumpserver requirements]# cd /opt/coco/

(py3) [root@jumpserver coco]# cp conf\_example.py conf.py

15 NAME = "coco"

18 CORE\_HOST = 'http://192.168.122.101:8080'

21 BIND\_HOST = '0.0.0.0'

24 SSHD\_PORT = 2222

40 LOG\_LEVEL = 'WARN'

4、启动cocod服务

(py3) [root@jumpserver coco]# ./cocod start -d

启动成功后去Jumpserver 会话管理-终端管理接受coco的注册

(py3) [root@jumpserver ~]# netstat -antp | grep :2222

tcp 0 0 0.0.0.0:2222 0.0.0.0:\* LISTEN 14640/python3

**测试ssh连接跳板机**

[root@localhost ~]# ssh martin@192.168.122.101 -p 2222

**安装luna提供web前端**

(py3) [root@jumpserver opt]# tar xf luna.tar.gz

(py3) [root@jumpserver opt]# chown -R root.root luna

**安装nginx**

(py3) [root@jumpserver opt]# yum install -y nginx

(py3) [root@jumpserver nginx]# vim /etc/nginx/nginx.conf

client\_max\_body\_size 100m; # 录像上传大小限制

location /luna/ {

try\_files $uri / /index.html;

alias /opt/luna/; # luna 路径，如果修改安装目录，此处需要修改

}

location /media/ {

add\_header Content-Encoding gzip;

root /opt/jumpserver/data/; # 录像位置，如果修改安装目录，此处需要修改

}

location /static/ {

root /opt/jumpserver/data/; # 静态资源，如果修改安装目录，此处需要修改

}

location /socket.io/ {

proxy\_pass http://localhost:5000/socket.io/; # 如果coco安装在别的服务器，请填写它的ip

proxy\_buffering off;

proxy\_http\_version 1.1;

proxy\_set\_header Upgrade $http\_upgrade;

proxy\_set\_header Connection "upgrade";

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header Host $host;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

access\_log off;

}

location /guacamole/ {

proxy\_pass http://localhost:8081/; # 如果guacamole安装在别的服务器，请填写它的ip

proxy\_buffering off;

proxy\_http\_version 1.1;

proxy\_set\_header Upgrade $http\_upgrade;

proxy\_set\_header Connection $http\_connection;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header Host $host;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

access\_log off;

}

location / {

proxy\_pass http://localhost:8080; # 如果jumpserver安装在别的服务器，请填写它的ip

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header Host $host;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

}

(py3) [root@jumpserver nginx]# nginx -t

(py3) [root@jumpserver nginx]# systemctl start nginx

(py3) [root@jumpserver nginx]# systemctl enable nginx