

Ansible Tower 安装配置及使用指南

Ansible Tower Version: 3.0.x

Quan Hu <qhu@redhat.com>

修订历史					
编号	日期	修订描述	版本	作者	审核
1	2016-12-20	架建初稿（安装部分）	V1.0	胡权	
2	2017-01-17	增加 Windows PowerShell 配置	V1.1	胡权	
3	2017-02-10	增加 Ansible Tower 配置和管理	V1.2	胡权	

目录

1.	前言	5
2.	格式约定	6
2.1	格式约定	6
3.	需求条件	7
3.1	支持的操作系统	7
3.2	硬件要求	7
3.3	软件依赖 (RHEL)	7
4.	环境介绍	8
4.1	安装介质	8
4.2	操作系统环境	8
4.3	/var 分区检查	8
4.4	YUM 源	8
4.5	License	9
5.	安装 ANSIBLE TOWER 在 RHEL6 上	10
5.1	安装介质准备	10
5.2	设置 Inventory 文件	10
5.3	安装 Tower	12
5.4	导入 License	12
5.5	访问 Ansible Tower	13
6.	安装 ANSIBLE TOWER 在 RHEL7 上	14
6.1	安装介质准备	14
6.2	设置 Inventory 文件	14
6.3	安装 Tower	16
6.4	导入 License	16
6.5	访问 Ansible Tower	17
7.	管理 ANSIBLE TOWER	18
7.1	Tower 用户密码修改	18
7.1.1	命令行修改	18
7.1.2	Webconsole 修改	18
7.2	Tower 服务管理	19
7.3	Organization 组织管理	19
7.3.1	创建新组织	20
7.3.2	管理组织	21
7.4	User 用户管理	22
7.4.1	创建新用户	22
7.4.2	添加用户到组织	23
7.5	Team 团队管理	25
7.5.1	创建新团队	26
7.5.2	添加用户到团队	27
7.6	Credential 凭证管理	29
7.6.1	创建凭证	29
7.6.2	凭证的类型及属性	31

7.6.2.1	Machine 凭证.....	31
7.6.2.2	Network 凭证.....	33
7.6.2.3	Source Control 凭证.....	33
7.6.2.4	Amazon Web Services 凭证.....	34
7.6.2.5	Rackspace 凭证.....	35
7.6.2.6	VMware vCenter 凭证.....	35
7.6.2.7	Red Hat Satellite 6 凭证.....	35
7.6.2.8	Red Hat CloudForms 凭证.....	36
7.6.2.9	Google Compute Engine 凭证.....	36
7.6.2.10	Microsoft Azure Classic (Deprecated) 凭证.....	37
7.6.2.11	Microsoft Azure Resource Manager 凭证.....	37
7.6.2.12	OpenStack 凭证.....	38
7.7	Project 项目管理.....	39
7.7.1	创建新项目.....	40
7.7.2	Playbook 剧本的管理方式.....	42
7.7.2.1	手动管理.....	42
7.7.2.2	SCM 源代码控制管理.....	42
7.8	Inventory 清单管理.....	43
7.8.1	创建 Inventory.....	44
7.8.2	在 Inventory 中添加 Host.....	45
7.8.3	在 Inventory 中添加 Group.....	46
7.8.4	在 Group 中添加子 Group 和 Host.....	47
7.8.4.1	添加新的子 GROUP 和 HOST 到 GROUP.....	48
7.8.4.2	移动或复制 GROUP 和 HOST 到 GROUP.....	48
7.9	Job Template 作业模板管理.....	53
7.9.1	创建新作业模板.....	53
7.9.2	运行作业.....	57
7.9.3	计划作业.....	60
7.9.3.1	创建一个计划作业.....	60
7.10	Job 作业管理.....	63
7.11	Permissions 权限管理.....	67
7.11.1	角色层次结构和继承.....	68
7.11.2	用户权限.....	69
7.11.3	组织权限.....	73
7.11.4	项目权限.....	79
7.11.5	清单权限.....	82
7.11.6	凭证权限.....	85
7.11.7	作业模板权限.....	88
7.11.8	只读权限.....	91
7.11.9	内置角色列表.....	91
8.	配置 WINDOWS 被 ANSIBLE 管理.....	93
8.1	Ansible 安装 winrm 模块.....	93
8.2	配置 Windows PowerShell.....	93
8.3	连接测试.....	95

1. 前言

Ansible 是一个开源的配置管理和业务流程工具。它可以自动化和标准化的配置远程主机和虚拟机。它的编排功能允许 Ansible 并列的启动和正常关闭多种应用程序。所以 Ansible 可以以零停机的方式执行多个系统的轧制更新。

Ansible Tower 是一个基于 Web 的用户界面，提供了 IT 自动化的企业解决方案。它有一个友好用户的仪表板来管理部署和监控资源。Ansible Tower 为 Ansible 增加自动化，可视化管理和监控能力。

本文档将分别介绍 Ansible Tower 在 RHEL6 和 RHEL7 上的安装。

2. 格式约定

2.1 格式约定

序号	格式开头	格式说明
1	以“#”开头	代表 root 用户；
2	以“\$”开头	代表普通用户；
3	以边框“灰色底纹”标注	表示命令或输入字符；
4	以边框“白色底纹”标注	表示配置文件配置项
5	以“红色”标注	表示特殊说明
6	以“楷体”标注	表示注释内容；

3. 需求条件

3.1 支持的操作系统

- Red Hat Enterprise Linux 6 64-bit
- Red Hat Enterprise Linux 7 64-bit
- CentOS 6 64-bit
- CentOS 7 64-bit
- Ubuntu 12.04 LTS 64-bit
- Ubuntu 14.04 LTS 64-bit

3.2 硬件要求

名称	最小值	推荐值
内存	2GB	4GB+ (100 forks), 16GB+ (400 forks)
磁盘	20GB (/var 分区 10GB)	40GB+ (/var 分区 20GB+)
CPU	2C	8C+

3.3 软件依赖 (RHEL)

操作系统版本	依赖
Red Hat Enterprise Linux 6	rhsc1 源
Red Hat Enterprise Linux 7	无

注：以上依赖源分别为 RHEL6 和 RHEL7 版本，官方文档中提到 RHEL7 依赖 extras 源，但是在实际安装过程中，并没有用到 extras 源中的软件包，所以 RHEL7 使用 ISO 中的软件包就可以成功完成安装。从 Ansible Tower 3.0 开始，Tower 在 RHEL6 上的运行环境是 python2.7，所以在 RHEL6 上安装 Tower 不在依赖 optional 源，而是依赖于 rhsc1 源。

4. 环境介绍

4.1 安装介质

推荐使用 Bundled 安装程序，从 Ansible Tower 2.3.0 开始，Tower 安装支持 bundled 安装程序，Bundled 安装程序虽然打包了大部分依赖软件包，但是仍然需要访问 Red Hat Enterprise Linux 源。

Bundled 下载地址：<https://releases.ansible.com/ansible-tower/setup-bundle/>，
下载最新的版本，当前最新版本是 3.0.3

ansible-tower-setup-bundle-3.0.3-1.el6.tar.gz
ansible-tower-setup-bundle-3.0.3-1.el7.tar.gz

4.2 操作系统环境

操作系统版本	IP 地址	主机名
Red Hat Enterprise Linux 6.8	172.168.0.10	ansible6.example.com
Red Hat Enterprise Linux 7.3	172.168.0.11	ansible7.example.com

注：本文档介绍的是 Ansible Tower 分别在 RHEL6 和 RHEL7 上的安装，实际环境中可根据需求选择一个版本即可。

4.3 /var 分区检查

安装 Ansible Tower，/var 分区至少有 10G 的可用空间，否则安装前检查会无法通过，
检查方法如下：

# df -hP /var/					
Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/mapper/VolGroup-lv_var	15G	114M	14G	1%	/var

注：如果/var 不是一个单独的文件系统，也可以使用上面的命令进行检查，df 命令会打印
/var 目录所在文件系统上的大小，例如：/分区。

4.4 YUM 源

Ansibel Tower Bundled 安装程序虽然包含了大部分依赖软件包，但是仍然需要额外的
源，除了基础源（系统光盘自带的 RPM 包）之外，Red Hat Enterprise Linux 6 还需要订

阅 rhsc1 源, Red Hat Enterprise Linux 7 不需要订阅额外的源。

RHEL6:

```
# yum repolist --noplugins
```

repo id	repo name	status
rhel-server-rhsc1-6-rpms	Red Hat Software Collections	4,973
rhel-6-server-rpms	Red Hat Enterprise Linux 6 Server	3,855

repolist: 8,828

RHEL7:

```
# yum repolist --noplugins
```

repo id	repo name	status
rhel-7-server-rpms	Red Hat Enterprise Linux 7 Server	4,751

repolist: 4,819

4.5 License

Ansible Tower 企业版需要收费,如需订阅,请联系红帽:<https://access.redhat.com>。

如果需要测试及评估 Ansible Tower, 可申请试用版基础 license, 基础 License 只支持 10 个节点, 申请地址: <http://ansible.com/license>; 红帽员工可以申请 50 个节点的企业版 License, 申请地址: https://store.ansible.com/redhat/tower_license/。

5. 安装 ANSIBLE TOWER 在 RHEL6 上

本小节介绍的是 Ansible Tower 在 RHEL6 上的标准安装，使用单节点，内置数据库。

5.1 安装介质准备

上传 Bundled 安装介质至服务器 `ansible6.example.com:/opt` 目录下：

```
# ls /opt/ansible-tower-setup-bundle-3.0.3-1.el6.tar.gz
/opt/ansible-tower-setup-bundle-3.0.3-1.el6.tar.gz
```

解压：

```
# cd /opt
# tar -xf /opt/ansible-tower-setup-bundle-3.0.3-1.el6.tar.gz
# ls
ansible.cfg  backup.yml  bundle  group_vars  install.yml  inventory
licenses  README.md  restore.yml  roles  setup.sh
```

5.2 设置 Inventory 文件

Ansible Tower 3.0 的安装省略了运行 `./configure` 的步骤，在执行安装之前，需要手动修改 `inventory` 文件，根据当前的环境情况，进行配置。以下是配置示例：

(1) 单节点内置数据库：

```
[primary]
localhost ansible_connection=local

[secondary]

[database]

[all:vars]
admin_password='password'      #使用实际的密码替换 password
redis_password='password'      #使用实际的密码替换 password

pg_host='',
pg_port='',

pg_database='awx',
pg_username='awx',
pg_password='password'          #使用实际的密码替换 password
```

(2) 双节点+已安装的外部数据库：

```
[primary]
```

```

node1 ansible_connection=local #使用实际的主机替换 node1

[secondary]
node2 #使用实际的主机替换 node2

[database]

[all:vars]
admin_password='password' #使用实际的密码替换 password
redis_password='password' #使用实际的密码替换 password

pg_host= 'database' #使用实际的主机替换 database
pg_port=' 5432' #使用实际的端口替换 5432

pg_database=' awx'
pg_username=' awx'
pg_password=' password' #使用实际的密码替换 password

```

(3) 双节点+未安装的外部数据库:

```

[primary]
node1 ansible_connection=local #使用实际的主机替换 node1

[secondary]
node2 #使用实际的主机替换 node2

[database]
database #使用实际的主机替换 database

[all:vars]
admin_password='password' #使用实际的密码替换 password
redis_password='password' #使用实际的密码替换 password

pg_host= 'database' #使用实际的主机替换 database
pg_port=' 5432' #使用实际的端口替换 5432

pg_database=' awx'
pg_username=' awx'
pg_password=' password' #使用实际的密码替换 password

```

本文档使用单节点内置数据库，所以 inventory 文件配置如下：

```

[primary]
localhost ansible_connection=local

[secondary]

```

```
[database]

[all:vars]
admin_password='redhat'
redis_password='redhat'

pg_host=''
pg_port=''

pg_database='awx'
pg_username='awx'
pg_password='redhat'
```

注：**redis** 密码不支持空格和这些特殊符号：**【@, :, -, \, /, #】**，另外，如果使用双节点的 Tower，则数据库必须使用外部数据库，且数据库节点必须是一台单独的节点，不能是 Tower 节点中的一台。

5.3 安装 Tower

在配置完 inventory 文件之后，就可以执行 setup.sh 开始安装，

```
# ./setup.sh
[warn] Will install bundled Ansible
.....
中间的输出内容省略
.....
PLAY RECAP *****
localhost           : ok=126  changed=53   unreachable=0    failed=0

The setup process completed successfully.
Setup log saved to /var/log/tower/setup-2016-12-20-00:04:42.log
```

最后看到以上输出，表示安装成功了。

注：如果使用低于 RHEL6.7 的版本，在安装过程中会提示找不到 PyYAML，则需要额外下载 PyYAML 包，因为 PyYAML 是在 RHEL6.7 才加入的光盘镜像中的。

5.4 导入 License

将申请的 License key 文件上传至 Ansible Tower 的 /etc/tower 目录下，并命名为 license，如下：

```
# ls /etc/tower/license
/etc/tower/license
```

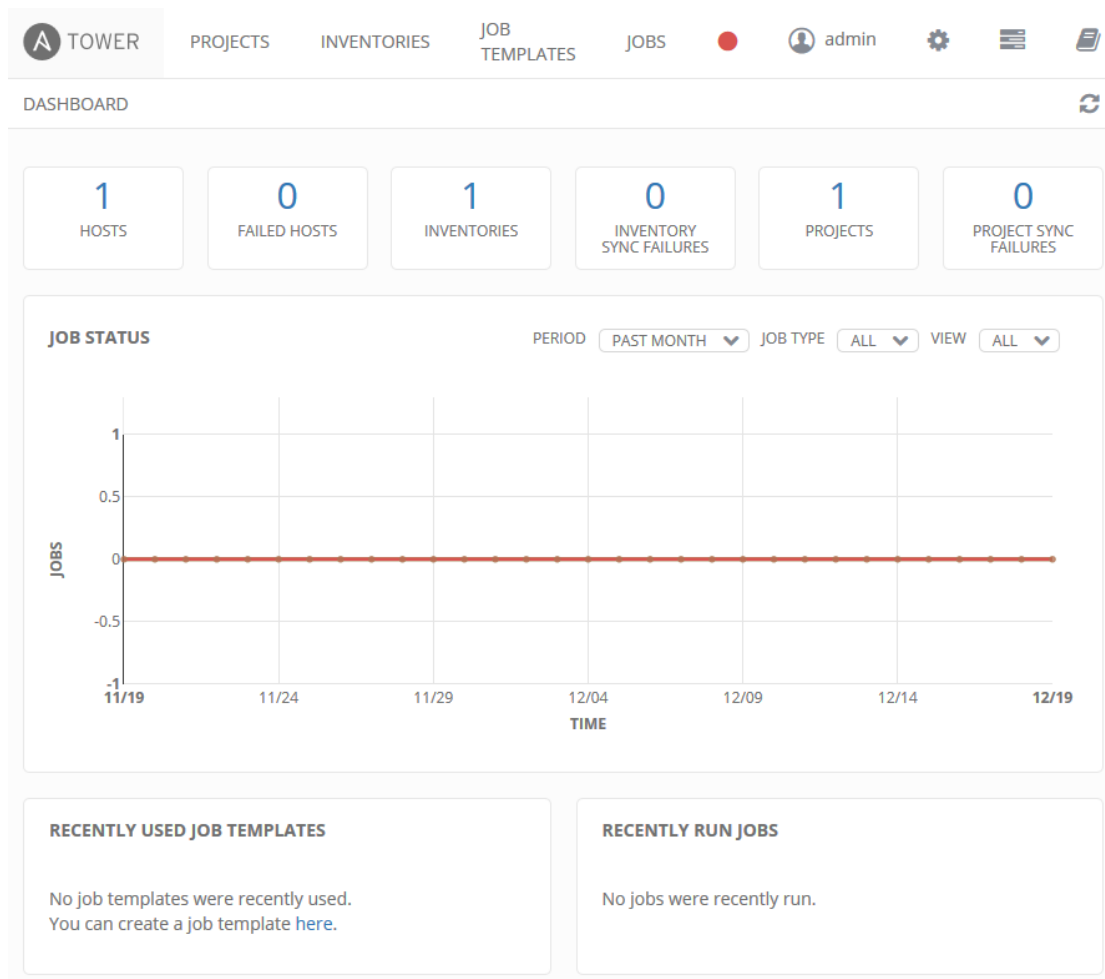
5.5 访问 Ansible Tower

通过浏览器访问：<https://172.168.0.10>，如下：



The login page for Ansible Tower. It features the Ansible Tower logo (a red 'A' in a circle) and the text "ANSIBLE TOWER by Red Hat". Below the logo, it says "Welcome to Ansible Tower! Please sign in." There are two input fields: "USERNAME" and "PASSWORD". A green "SIGN IN" button is located at the bottom right of the form.

使用用户名 admin 及 inventory 文件中设置的密码进行登陆，登陆后界面如下：



6. 安装 ANSIBLE TOWER 在 RHEL7 上

本小节介绍的是 Ansible Tower 在 RHEL7 上的标准安装，使用单节点，内置数据库。

6.1 安装介质准备

上传 Bundled 安装介质至服务器 `ansible7.example.com:/opt` 目录下：

```
# ls /opt/ansible-tower-setup-bundle-3.0.3-1.el7.tar.gz
/opt/ansible-tower-setup-bundle-3.0.3-1.el7.tar.gz
```

解压：

```
# cd /opt
# tar -xf /opt/ansible-tower-setup-bundle-3.0.3-1.el7.tar.gz
# ls
ansible.cfg  backup.yml  bundle  group_vars  install.yml  inventory
licenses  README.md  restore.yml  roles  setup.sh
```

6.2 设置 Inventory 文件

Ansible Tower 3.0 的安装省略了运行 `./configure` 的步骤，在执行安装之前，需要手动修改 `inventory` 文件，根据当前的环境情况，进行配置。以下是配置示例：

(1) 单节点内置数据库：

```
[primary]
localhost ansible_connection=local

[secondary]

[database]

[all:vars]
admin_password='password'      #使用实际的密码替换 password
redis_password='password'      #使用实际的密码替换 password

pg_host='',
pg_port='',

pg_database='awx'
pg_username='awx'
pg_password='password'         #使用实际的密码替换 password
```

(2) 双节点+已安装的外部数据库：

```
[primary]
```

```

node1 ansible_connection=local #使用实际的主机替换 node1

[secondary]
node2 #使用实际的主机替换 node2

[database]

[all:vars]
admin_password='password' #使用实际的密码替换 password
redis_password='password' #使用实际的密码替换 password

pg_host='database' #使用实际的主机替换 database
pg_port='5432' #使用实际的端口替换 5432

pg_database='awx'
pg_username='awx'
pg_password='password' #使用实际的密码替换 password

```

(3) 双节点+未安装的外部数据库:

```

[primary]
node1 ansible_connection=local #使用实际的主机替换 node1

[secondary]
node2 #使用实际的主机替换 node2

[database]
database #使用实际的主机替换 database

[all:vars]
admin_password='password' #使用实际的密码替换 password
redis_password='password' #使用实际的密码替换 password

pg_host='database' #使用实际的主机替换 database
pg_port='5432' #使用实际的端口替换 5432

pg_database='awx'
pg_username='awx'
pg_password='password' #使用实际的密码替换 password

```

本文档使用单节点内置数据库，所以 inventory 文件配置如下:

```

[primary]
localhost ansible_connection=local

[secondary]

```

```
[database]

[all:vars]
admin_password='redhat'
redis_password='redhat'

pg_host='',
pg_port='',

pg_database='awx'
pg_username='awx'
pg_password='redhat'
```

注：**redis** 密码不支持空格和这些特殊符号：**【@, :, -, \, /, #】**，另外，如果使用双节点的 **Tower**，则数据库必须使用外部数据库，且数据库节点必须是一台单独的节点，不能是 **Tower** 节点中的一台。

6.3 安装 Tower

在配置完 `inventory` 文件之后，就可以执行 `setup.sh` 开始安装，

```
# ./setup.sh
[warn] Will install bundled Ansible
.....
中间的输出内容省略
.....
PLAY RECAP *****
localhost           : ok=122  changed=52   unreachable=0    failed=0

The setup process completed successfully.
Setup log saved to /var/log/tower/setup-2016-12-19-16:26:34.log
```

最后看到以上输出，表示安装成功了。

6.4 导入 License

将申请的 `License key` 文件上传至 `Ansible Tower` 的 `/etc/tower` 目录下，并命令为 `license`，如下：

```
# ls /etc/tower/license
/etc/tower/license
```

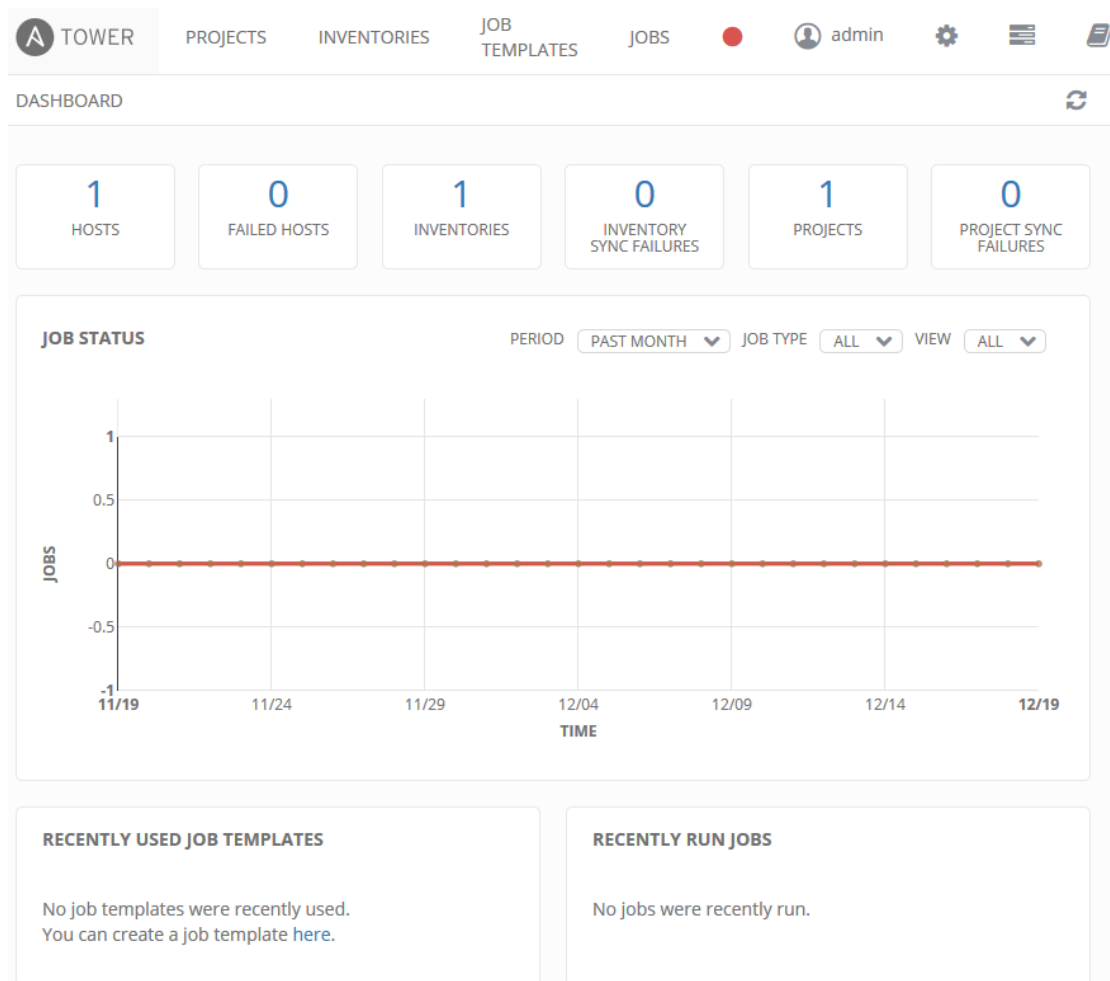

6.5 访问 Ansible Tower

通过浏览器访问：<https://172.168.0.11>，如下：



The login page for Ansible Tower. It features the Ansible Tower logo (a red 'A' in a circle) and the text "ANSIBLE TOWER by Red Hat". Below the logo, it says "Welcome to Ansible Tower! Please sign in." There are two input fields: "USERNAME" and "PASSWORD". A green "SIGN IN" button is located at the bottom right of the form.

使用用户名 admin 及 inventory 文件中设置的密码进行登陆，登陆后界面如下：



7. 管理 ANSIBLE TOWER

7.1 Tower 用户密码修改

7.1.1 命令行修改

Ansible Tower 用户密码修改使用 `tower-manage` 命令修改，例如：修改 admin 密码

```
# tower-manage changepassword admin
Changing password for user 'admin'
Password:          #输入密码
Password (again):  #再次输入密码
Password changed successfully for user 'admin'
```

7.1.2 Webconsole 修改

使用 admin 用户登录 Ansible Tower，然后点击右上角的“Setting”→“USERS”→“admin”，如下：

The screenshot shows the Ansible Tower web interface. At the top, there's a navigation bar with 'TOWER', 'PROJECTS', 'INVENTORIES', 'JOB TEMPLATES', 'JOBS', and a user profile 'admin'. Below this is a breadcrumb 'SETTINGS / USERS / ADMIN'. The main content area is titled 'ADMIN' and has a tabbed interface. The 'DETAILS' tab is selected, showing a form for user information. The form includes fields for 'FIRST NAME', 'LAST NAME', 'EMAIL' (pre-filled with 'admin@example.com'), 'USERNAME' (pre-filled with 'admin'), 'PASSWORD' (with a 'SHOW' button), and 'CONFIRM PASSWORD' (with a 'SHOW' button). There's also a 'USER TYPE' dropdown menu set to 'System Administrator'. At the bottom right, there are 'CANCEL' and 'SAVE' buttons.

第一次使用 webconsole 修改 admin 密码，需要完善 admin 的用户信息，*都是必填的，输入新的密码，点击右下方的“SAVE”保存即可。

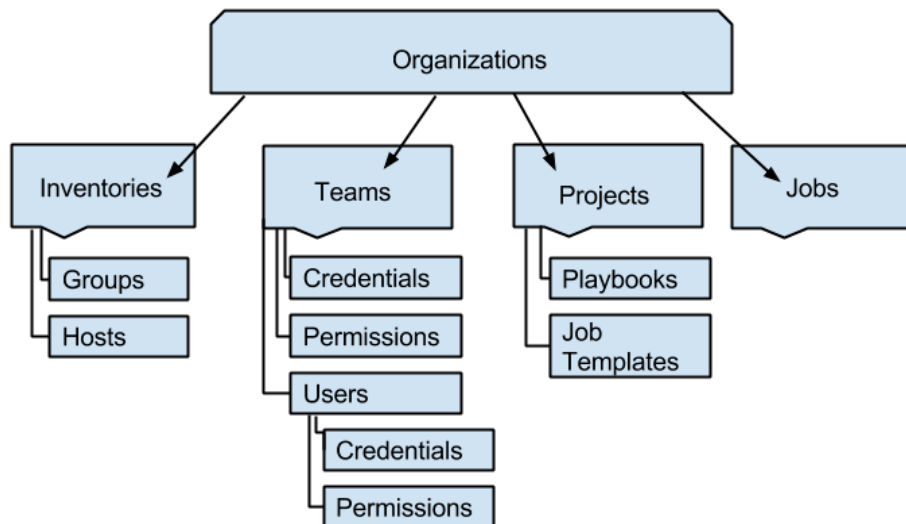
7.2 Tower 服务管理

Ansible Tower 服务的启停使用 `ansible-tower-service` 命令，示例如下：

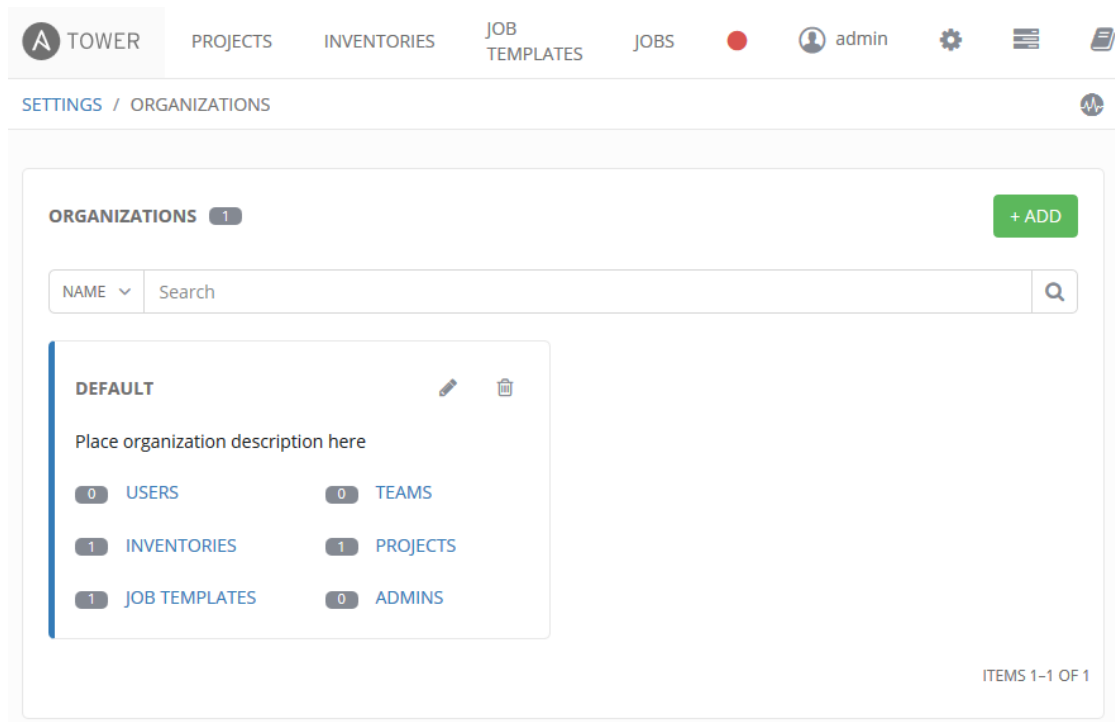
```
# ansible-tower-service stop      #停止 tower
# ansible-tower-service start     #启动 tower
# ansible-tower-service status    #查看 tower 状态
```

7.3 Organization 组织管理

组织是一个用户、团队、项目和清单的逻辑集合，位于 Tower 层次结构中的最高级，如下图所示：

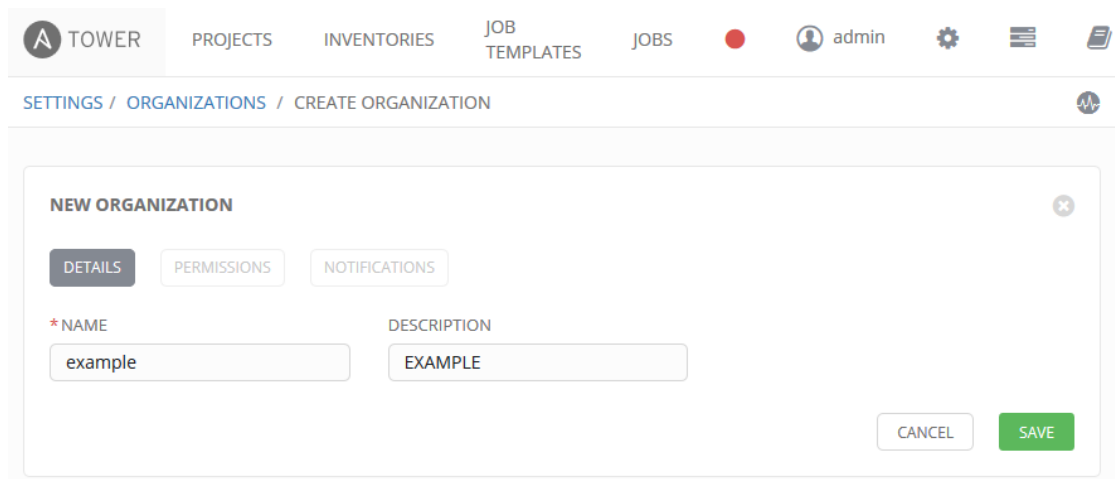


安装完 Ansible Tower 后，会有一个默认的组织，点击右上角的“Setting” → “ORGANIZATION”，即可看到默认 ORGANIZATION: default，如下：

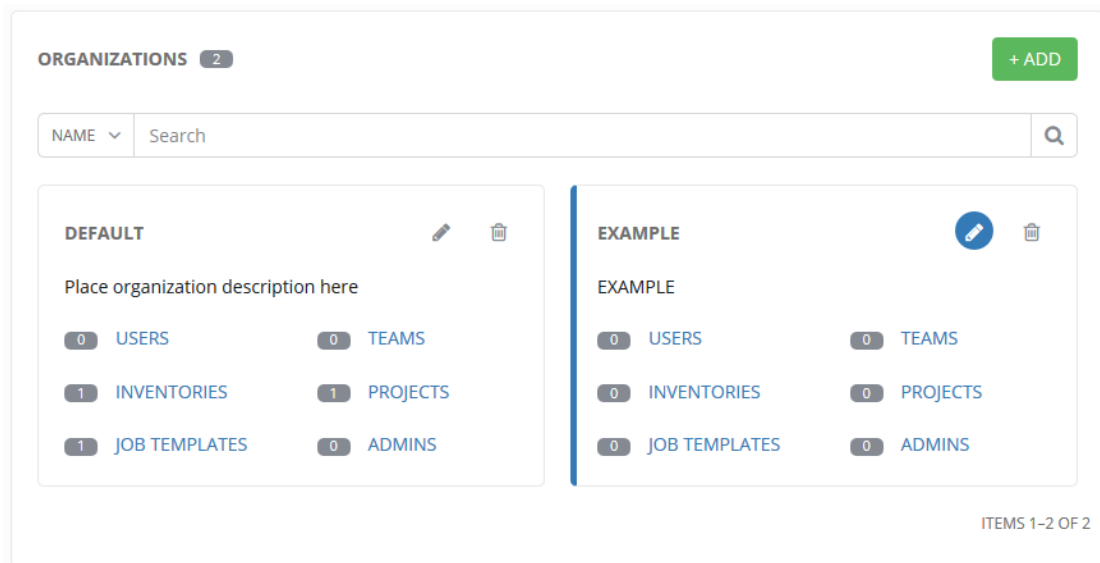


7.3.1 创建新组织

点击“ORGANIZATION”后面的“+ADD”，输入组织名称以及描述，如下：



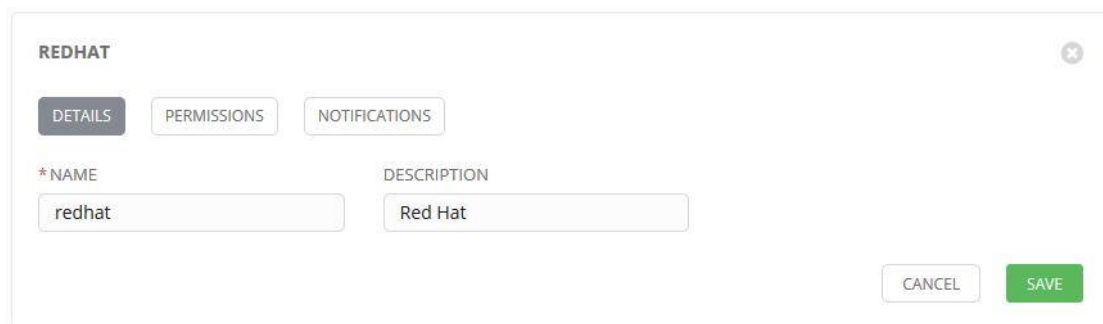
点击“SAVE”保存，就可以看到有两个 ORGANIZATION 了，如下：



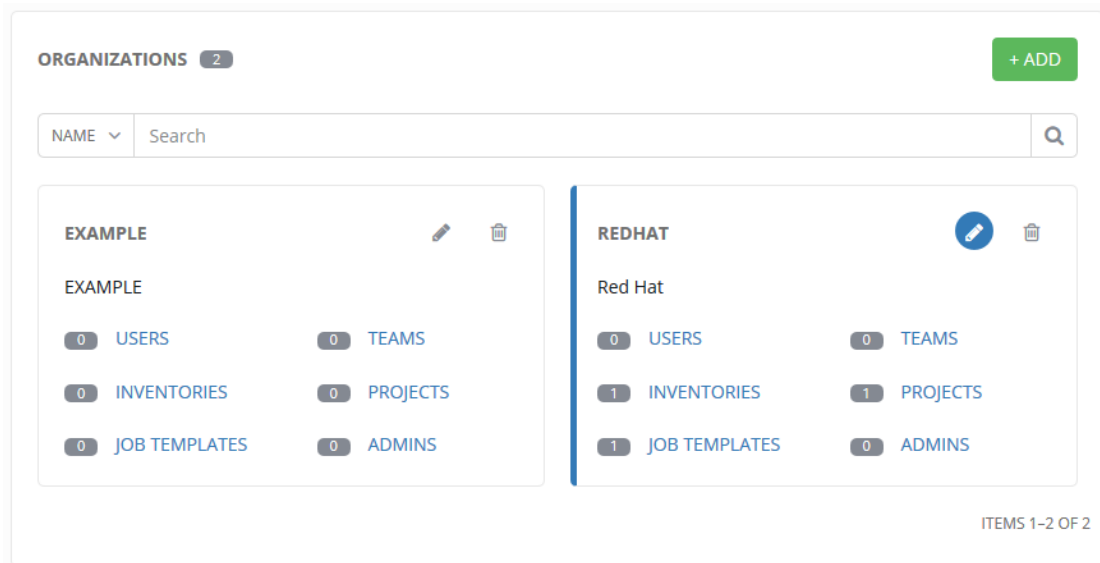
注：如果 ansible tower 使用的 Basic license，则只能使用默认 ORGANIZATION，只有企业版和付费版的 licenses 才有权利创建新的 ORGANIZATION。

7.3.2 管理组织

默认组织 default 可以进行修改，点击 default 后面的按钮进行编辑，如下：



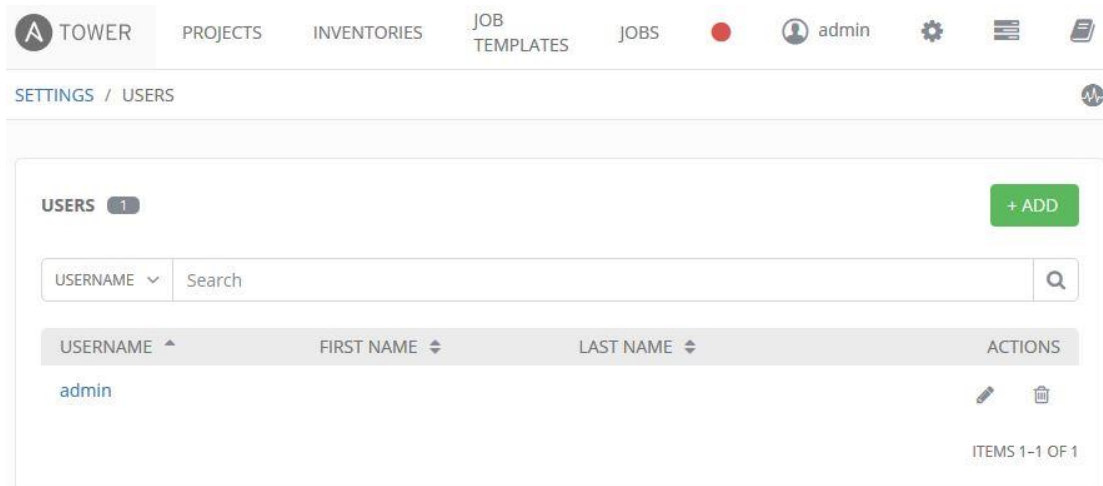
修改名称，添加描述，点击“SAVE”保存，如下：



依次点击组织名称下的“USERS”、“TEAMS”、“INVENTORIES”、“PROJECTS”、“JOB TEMPLATES”和“ADMINS”可以查看组织中的用户、团队、清单、项目、作业模板和管理员。

7.4 User 用户管理

安装完 Ansible Tower 后，默认只有一个用户 admin，点击右上角的“Setting”→“USERS”，即可看到默认用户 admin，如下：



7.4.1 创建新用户

点击“+ADD”添加新的用户，如下：

NEW USER

ADMIN

DETAILS

ORGANIZATIONS

TEAMS

GRANTED PERMISSIONS

* FIRST NAME

Red

* LAST NAME

Hat

* EMAIL

redhat@example.com

* USERNAME

redhat

* ORGANIZATION

Q

example

* PASSWORD

SHOW

.....

* CONFIRM PASSWORD

SHOW

.....

USER TYPE

System Administrator

▼

CANCEL

SAVE

输入用户信息，点击“SAVE”保存，如下：

USERS

2

+ ADD

USERNAME ▼

Search

Q

USERNAME ▲	FIRST NAME ⇅	LAST NAME ⇅	ACTIONS
admin			<div><div></div><div></div></div>
redhat	Red	Hat	<div><div></div><div></div></div>

ITEMS 1-2 OF 2

注：用户类型可以选择“Normal User”，“System Administrator”和“System Auditor”。

Normal User：Normal User 被赋予了角色和权利后，对 inventory 和 projects 具有有限的读写访问权限，

System Auditor：审计员隐藏继承了对所有对象的只读权限。

System Administrator：系统管理员在整个 tower 环境中具有管理，读写权限。

7.4.2 添加用户到组织

为 ORGANIZATION 添加用户，点击右上角的“Setting” → “ORGANIZATION”，如下：

ORGANIZATIONS 2

+ ADD

NAME

Search

Q

EXAMPLE

EXAMPLE

4

USERS

1

TEAMS

0

INVENTORIES

0

PROJECTS

0

JOB TEMPLATES

0

ADMINS

REDHAT

Red Hat

0

USERS

0

TEAMS

1

INVENTORIES

1

PROJECTS

1

JOB TEMPLATES

0

ADMINS

ITEMS 1-2 OF 2

点击组织“REDHAT”下面的“USERS”，如下：

REDHAT | USERS 0

+ ADD USER

PLEASE ADD ITEMS TO THIS LIST

点击“+ADD”，添加新的用户，如下：

REDHAT | ADD USERS

USERNAME

Search

Q

	USERNAME	FIRST NAME	LAST NAME
<input type="checkbox"/>	admin		
<input type="checkbox"/>	jone	Jone	Wu
<input type="checkbox"/>	lisi	Li	Si
<input type="checkbox"/>	redhat	Red	Hat
<input type="checkbox"/>	zhangsan	Zhang	San

ITEMS 1-5 OF 5

CANCEL

SAVE

选择需要添加的用户，点击“SAVE”保存，如下：

REDHAT | USERS 4

USERNAME

Search

Q

+ ADD USER

USERNAME	FIRST NAME	LAST NAME	ACTIONS
jone	Jone	Wu	<input type="checkbox"/> <input type="checkbox"/>
lisi	Li	Si	<input type="checkbox"/> <input type="checkbox"/>
redhat	Red	Hat	<input type="checkbox"/> <input type="checkbox"/>
zhangsan	Zhang	San	<input type="checkbox"/> <input type="checkbox"/>

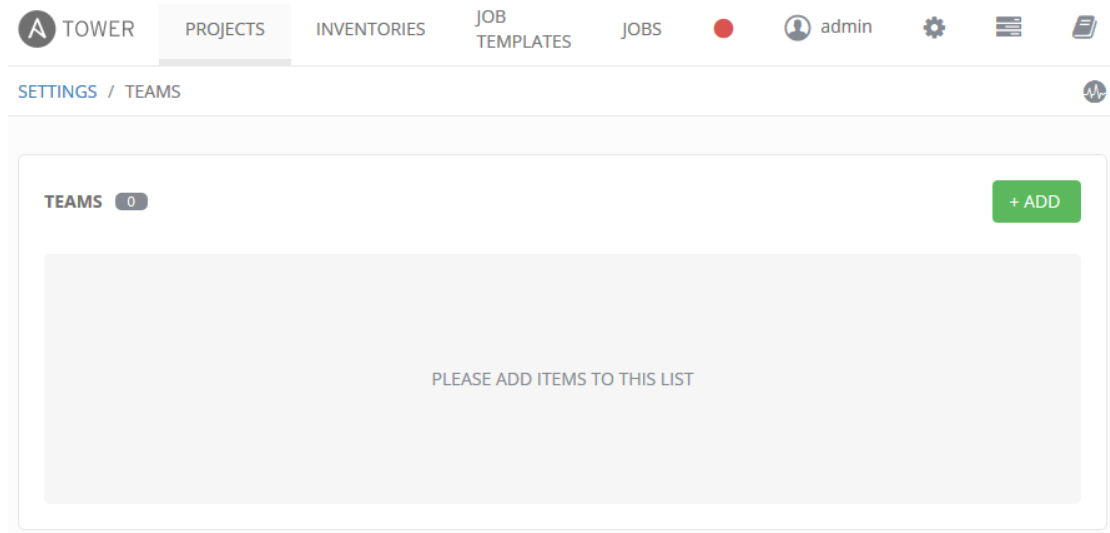
ITEMS 1-4 OF 4

7.5 Team 团队管理

团队是组织的一部分，与用户、项目、凭证和权限相关联，团队提供了一种方式来实现基于角色的访问控制方案和跨组织的责任委派，权限可以授予整个团队，而不是团队中的每一个用户。在组织中可以创建多个团队，对每个团队分配权限，与对用户分配权限一样。

7.5.1 创建新团队

安装完 Ansible Tower 后，默认是没有团队的，点击右上角的“Setting” → “TEAMS”，如下：



点击“+ADD”，创建新的团队，如下：

NEW TEAM

DETAILS USERS GRANTED PERMISSIONS

* NAME: development DESCRIPTION: Research and Development Dep. * ORGANIZATION: example

CANCEL SAVE

输入团队的信息，点击“SAVE”保存，然后点击“USERS”，如下：

DEVELOPMENT

DETAILS USERS GRANTED PERMISSIONS

USER Search + ADD

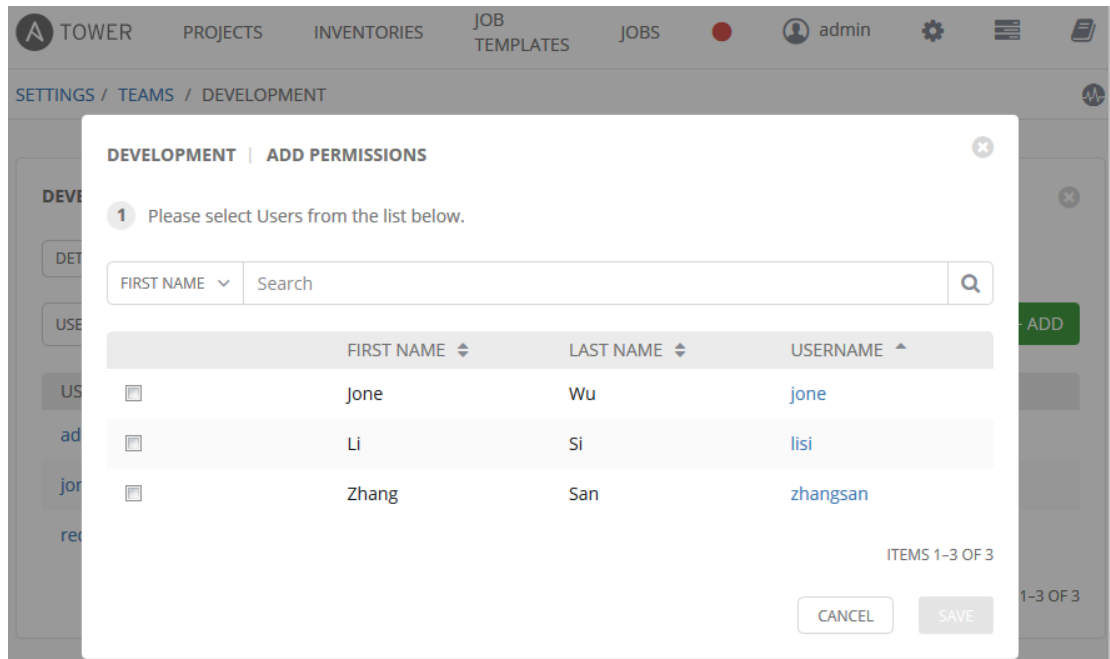
USER	ROLE
admin	SYSTEM ADMINISTRATOR
jone	SYSTEM AUDITOR
redhat	SYSTEM ADMINISTRATOR

ITEMS 1-3 OF 3

可以看到 development 这个团队中已经有了三个用户，其中 admin 和 redhat 是系统管理员，jone 是审计员。默认情况下，管理员对团队都有有限的权限。

7.5.2 添加用户到团队

在组“development”的显示页面，点击“USERS”→“+ADD”添加其他用户，如下：



勾选需要添加的用户。

注：jone 用户在这里可以再次勾选，是因为 jone 除了审计权限外，也可以拥有普通用户的权限。

为用户选择对应的 ROLES（Admin 或者 Member），不选则只有只读权限，如下：

DEVELOPMENT | ADD PERMISSIONS

1

Please select Users from the list below.

FIRST NAME

Search

Q

	FIRST NAME	LAST NAME	USERNAME
<input type="checkbox"/>	Jone	Wu	jone
<input checked="" type="checkbox"/>	Li	Si	lisi
<input checked="" type="checkbox"/>	Zhang	San	zhangsan

ITEMS 1-3 OF 3

2

Please assign roles to the selected users/teams

KEY

Li Si

USER

Member

X

Zhang San

USER

Admin

X

CANCEL

SAVE

注：一个用户可以授予多个 roles，默认权限为 read（只读）。

Admin： 用户将对组有所有的权限

Member： 用户是组里的一个成员

点击“SAVE”保存，如下：

DEVELOPMENT

DETAILS

USERS

GRANTED PERMISSIONS

USER

Search

Q

+ ADD

USER	ROLE
admin	SYSTEM ADMINISTRATOR
jone	SYSTEM AUDITOR
lisi	MEMBER
redhat	SYSTEM ADMINISTRATOR
zhangsan	ADMIN

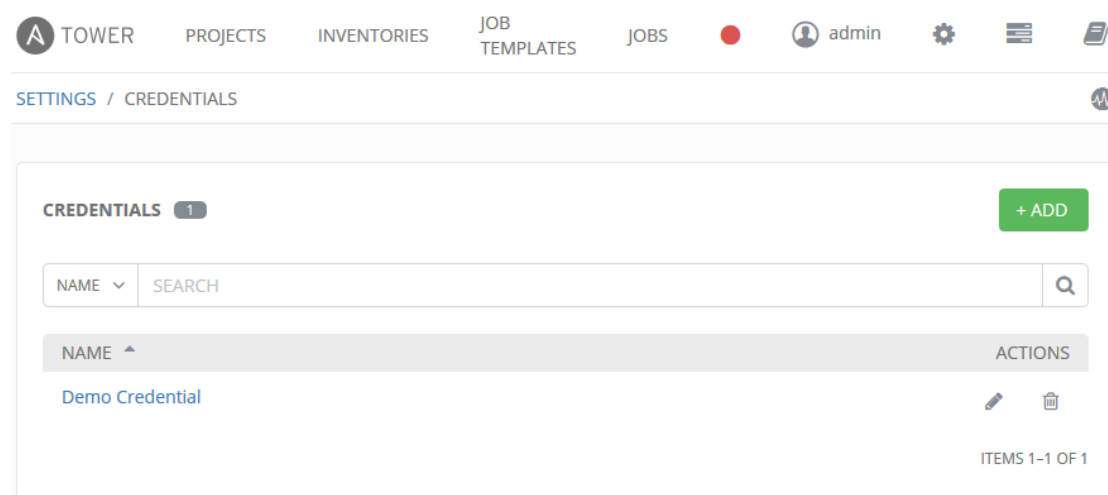
ITEMS 1-5 OF 5

7.6 Credential 凭证管理

Tower 在对主机运行 Job 的时候需要使用 Credentials (凭证) 认证, 凭证导入到 Tower 后将会被加密, 任何用户在命令行都无法取得文本内容, 一旦密码或者密钥写入 Tower 接口, 他将会被加密存入 tower 数据库, 从 tower 上是无法取得的, 你可以授予用户或者组使用凭证的权利, 但是并没有对用户公开凭证, 如果有一个用户离开了组织或者移动到另一个组, 你不必更新系统中的所有密钥, 因为凭证在 tower 里面。

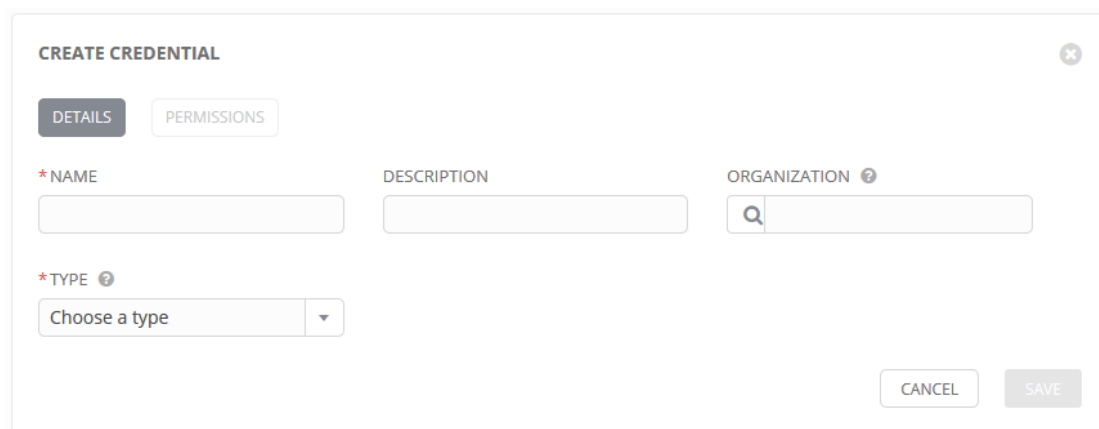
7.6.1 创建凭证

安装完 Ansible Tower 后, 默认有一个“Demo Credential”, 点击右上角的“Setting” → “CREDENTIALS”, 如下:



创建一个使用 SSH 公钥认证的凭证, 操作如下:

点击 “+ADD”, 创建新的凭证, 如下:



输入凭证的信息, 如下:

CREATE CREDENTIAL

DETAILS

PERMISSIONS

*NAME

SSH

DESCRIPTION

ssh public key

ORGANIZATION ?

example

*TYPE ?

Machine

TYPE DETAILS

USERNAME

PASSWORD

SHOW

☐ Ask at runtime?

PRIVATE KEY PASSPHRASE

SHOW

☐ Ask at runtime?

PRIVILEGE ESCALATION ?

Choose a privilege escalation

VAULT PASSWORD

SHOW

☐ Ask at runtime?

PRIVATE KEY ?

```

-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEAoit1gUnZHZ0Qzn0EygjU36XeFFYqJt9BKOn6w1D1RDT5ADEd
6gmofR21LunEtLuq16xfUX3pjwDDmAxSCgjfB5L04+1VUziAkMi5t2pQ5xXns1W
D923o93BftcttHZe+iKCLt2rMNkxZjrvpJ/eUORxTkyXWz305cuqF2Kf3XeR90Hdb
UTmF6+/XHqe9C+BRUF495rXpojUh3AumyGV+12biQw27wW81WDqXeQowkd1BMu+
wnLVDLbyQs8VjfvxdpM90Ha0wj9ZaZCGDBbhV2Bk33hS2nvjt/PF82IIwc6yV5F4
KZZ/YoJ21hIvSF9M2tff3/LosZQYe9qZ+T1VHQIBIwKCAQEA1ET9uAkBB5Hjeu1G
NQ9jiqZHYwzzVr19ZzxSm7e7rBMsZPmdLb+v/Vzuy85qs97H2yFBNJBggr7etubJ
eNQ9VDAK0A5cpcKiWd1hvSJu3rrTxGB0r2ug1dIP/vCeRvEj0xZxHL1qQEX00VtY
1BW01SE2d+gU2vY15Wi041e916rhMUVgWHOF4IQGAx0hWVxfvojxR7J9MdfHXROU
yY2tau8YKqMwQKvncUCCx9SNCox7wX4fQqImlwgo+h+c13CzZjdgV1GDbL53g1Q

```

CANCEL

SAVE

带“*”为必填内容，凭证的类型根据实际需求进行选择，不同的类型需要填写内容也会不一样，点击“SAVE”保存，保存之后会看到私钥的内容发生了变化，这是因为私钥已经被加密了，如下：

PRIVATE KEY ?

\$encrypted\$

7.6.2 凭证的类型及属性

现在 Ansible Tower 支持的凭证类型有：

- Machine
- Network
- Source Control
- Amazon Web Services
- Rackspace
- VMware vCenter
- Red Hat Satellite 6
- Red Hat CloudForms
- Google Compute Engine
- Microsoft Azure Classic (Deprecated)
- Microsoft Azure Resource Manager
- OpenStack

7.6.2.1 Machine 凭证

Machine 凭证是 Tower 通过调用 Ansible，就像在命令行使用 Ansible 一样，指定 SSH 用户名，使用密码、SSH KEY、SSH KEY 的密钥或者 Tower 提示输入密码的方式进行连接，同时也支持普通用户进行特权提升来执行 playbook。

*TYPE ?

Machine ▼

TYPE DETAILS

USERNAME

PASSWORD

SHOW

☐ Ask at runtime?

PRIVATE KEY PASSPHRASE

SHOW

☐ Ask at runtime?

PRIVILEGE ESCALATION ?

Sudo ▼

PRIVILEGE ESCALATION USERNAME

PRIVILEGE ESCALATION PASSWORD

SHOW

☐ Ask at runtime?

VAULT PASSWORD

SHOW

☐ Ask at runtime?

PRIVATE KEY ?

Machine 凭证的属性：

- Username：使用 SSH 认证的用户。
- Password：使用 SSH 认证的用户的密码，密码会加密后存入 Tower 数据库，另外也可以勾选 “Ask at runtime?”，让 Tower 在需要的时候提示输入密码。
- Private Key：SSH KEY 私钥，KEY 会加密后存入 Tower 数据库。
- Private Key Passphrase：SSH KEY 私钥密码，密码会加密后存入 Tower 数据库，另外也可以勾选 “Ask at runtime?”，让 Tower 在需要的时候提示输入密码。
- Privilege Escalation：指定特权升级的类型，使用普通用户进行 SSH 连接，升级到特权用户执行 Playbook 时需要用到，等同于`--become-method=BECOME_METHOD`参数，可以指定的类型包括：`sudo` | `su` | `pbrun` | `pfexec`。
- Privilege Escalation Username：远程系统上进行特权升级后的用户，如 `root`
- Privilege Escalation Password：进行特权升级时提供的密码，如果是 `sudo`，则是 SSH 认证的用户的密码，`su` 则是提权升级后用户的密码。密码会加密后存入 Tower 数据库，另外也可以勾选 “Ask at runtime?”，让 Tower 在需要的时候提示输入

密码。

- Vault Password: 如果 Playbook 使用了 Ansible 加密模块 vault, 则需要指定 vault 密码, 另外也可以勾选 “Ask at runtime?”, 让 Tower 在需要的时候提示输入密码。

说明: 在使用计划任务调度作业的时候, 不能使用 “Ask at runtime?”。

7.6.2.2 Network 凭证


Network 凭证是使用 Ansible 网络模块连接和管理网络设备。

TYPE DETAILS

* USERNAME	PASSWORD	<input checked="" type="checkbox"/> AUTHORIZE
<input type="text"/>	<input type="button" value="SHOW"/> <input type="text"/>	

AUTHORIZE PASSWORD

<input type="button" value="SHOW"/>	<input type="text"/>
-------------------------------------	----------------------

SSH KEY 

Network 凭证的属性:

- Username: 连接网络设备的用户名
- Password: 连接网络设备的用户密码
- SSH Key: 通过 SSH 连接网络设备的 SSH KEY 私钥
- Authorize: 勾选这个选项后, 可以增加一个 RSA key 的授权密码。

7.6.2.3 Source Control 凭证

SCM (source control) 凭证用于项目克隆和从远程版本控制系统(git,svn 或 Mercurial)更新本地源代码库。

* TYPE ?

Source Control ▼

TYPE DETAILS

USERNAME	PASSWORD	PRIVATE KEY PASSPHRASE
<input type="text"/>	<input type="button" value="SHOW"/> <input type="text"/>	<input type="button" value="SHOW"/> <input type="text"/>

SCM PRIVATE KEY ?

Source Control 凭证的属性：

- Username：连接远程控制系统的用户名
- Password：连接远程控制系统的用户密码
- SCM Private Key：通过 SSH 连接远程控制系统的 SSH KEY 私钥
- Private Key Passphrase：SSH KEY 私钥的密码，如果 SSH KEY 私钥启用了加密，需要在这里指定私钥的密码。

注：Source Control 凭证不能配置 “Ask at runtime?”。

7.6.2.4 Amazon Web Services 凭证

该凭证用于与 Amazon Web Services 进行云 Inventory 同步。

* TYPE ?

Amazon Web Services ▼

TYPE DETAILS

* ACCESS KEY	* SECRET KEY	STS TOKEN ?
<input type="text"/>	<input type="button" value="SHOW"/> <input type="text"/>	<input type="button" value="SHOW"/> <input type="text"/>

传统的 Amazon Web Services 凭证由 AWS Access Key 和 Secret Key 组成。

Ansible Tower 2.4.0 支持 EC2 STS tokens（也被称为 IAM STS 凭证）安全令牌服务 (STS) 是一个 web 服务，能够为用户访问和管理 AWS 请求临时的，有效的权限凭证。

7.6.2.5 Rackspace 凭证

该凭证用于与 Rackspace 进行云 Inventory 同步。

*TYPE ?

Rackspace ▼

TYPE DETAILS

*USERNAME

*API KEY

SHOW

Rackspace 凭证由 Rackspace Username 和 API Key 组成。

7.6.2.6 VMware vCenter 凭证

该凭证用于与 VMware vCenter 进行 Inventory 同步。

*TYPE ?

VMware vCenter ▼

TYPE DETAILS

*VCENTER HOST ?

*USERNAME

*PASSWORD

SHOW

VMware vCenter 凭证的属性：

- vCenter Host：需要连接的 vCenter 主机名或者 IP 地址
- Username：连接 vCenter 使用的用户名
- Password：连接 vCenter 使用的用户密码

7.6.2.7 Red Hat Satellite 6 凭证

该凭证用于与 Red Hat Satellite 6 进行云 Inventory 同步。

*TYPE ?

Red Hat Satellite 6 ▼

TYPE DETAILS

*SATELLITE 6 HOST ?

*USERNAME

*PASSWORD

SHOW

Red Hat Satellite 6 凭证的属性：

- Satellite 6 Host：需要连接的 Satellite 6 主机名或者 IP 地址
- Username：连接 Satellite 6 使用的用户名
- Password：连接 Satellite 6 使用的用户密码

7.6.2.8 Red Hat CloudForms 凭证

该凭证用于与 Red Hat CloudForms 进行云 Inventory 同步。

* TYPE ?

Red Hat CloudForms ▼

TYPE DETAILS

* CLOUDFORMS HOST ? * USERNAME * PASSWORD

Red Hat CloudForms 凭证的属性：

- CloudForms Host：需要连接的 CloudForms 主机名或者 IP 地址
- Username：连接 CloudForms 使用的用户名
- Password：连接 CloudForms 使用的用户密码

7.6.2.9 Google Compute Engine 凭证

该凭证用于与 Google Compute Engine 进行云 Inventory 同步。

* TYPE ?

Google Compute Engine ▼

TYPE DETAILS

* SERVICE ACCOUNT EMAIL ADDRESS ? PROJECT ?

* RSA PRIVATE KEY ?

Google Compute Engine 凭证的属性：

- Service Account Email Address：Google Compute Engine 服务账号的邮箱地址
- RSA Private Key：服务账号的 PEM 文件
- Project：GCE 身份标识，它由两个单词加一个 3 位数字组成

7.6.2.10 Microsoft Azure Classic (Deprecated) 凭证

该凭证用于与 Microsoft Azure Classic 进行云 Inventory 同步。

*TYPE ?

Microsoft Azure Classic (depr.. ▼

TYPE DETAILS

*SUBSCRIPTION ID ?

*MANAGEMENT CERTIFICATE ?

Microsoft Azure Classic 凭证的属性：

- Subscription ID: Microsoft Azure Classic 账户的订阅 UUID
- Management Certificate: 上传至 Microsoft Azure Classic 控制台的证书对应的 PEM 文件

7.6.2.11 Microsoft Azure Resource Manager 凭证

该凭证用于与 Microsoft Azure Resource Manager 进行云 Inventory 同步。

*TYPE ?

Microsoft Azure Resource Ma. ▼

TYPE DETAILS

*SUBSCRIPTION ID ?

USERNAME

PASSWORD

SHOW

CLIENT ID

CLIENT SECRET

SHOW

TENANT ID

Microsoft Azure Resource Manager 凭证的属性：

- Subscription ID: Microsoft Azure 账户的订阅 UUID
- Username: 用于连接 Microsoft Azure 的用户名
- Password: 用于连接 Microsoft Azure 的用户密码

- Client ID: Microsoft Azure 账户的客户端 ID
- Client Secret: Microsoft Azure 账户的客户端密码
- Tenant ID: Microsoft Azure 账户的租客 ID

定义以下变量，传递服务主要凭证：

```
AZURE_CLIENT_ID
AZURE_SECRET
AZURE_SUBSCRIPTION_ID
AZURE_TENANT
```

定义以下变量，传递活动目录的用户名/密钥：

```
AZURE_AD_USER
AZURE_PASSWORD
AZURE_SUBSCRIPTION_ID
```

也可以将凭证作为参数传递给 Playbook 中的一个任务，优先顺序是：参数，环境变量，最后是家目录中的文件。

使用以下的参数作为服务的主要凭证传递给一个任务：

```
client_id
secret
subscription_id
tenant
```

或者传递以下参数作为活动目录的用户名/密码：

```
ad_user
password
subscription_id
```

7. 6. 2. 12 OpenStack 凭证

该凭证用于与 OpenStack 进行云 Inventory 同步。

* TYPE ?

OpenStack ▼

TYPE DETAILS

* HOST (AUTHENTICATION URL) ?

* USERNAME

* PASSWORD (API KEY)

SHOW

* PROJECT (TENANT NAME) ?

DOMAIN NAME ?

OpenStack 凭证的属性：

- Host (Authentication URL)：用于认证的主机

- Username: 连接 OpenStack 使用的用户名
- Password (API Key): 连接 OpenStack 使用的用户密码或者 API key
- Project (Tenet Name): OpenStack 的租户名称或者 ID, 这个值通常与 Username 相同
- Domain name: 连接 OpenStack 使用的 FQDN

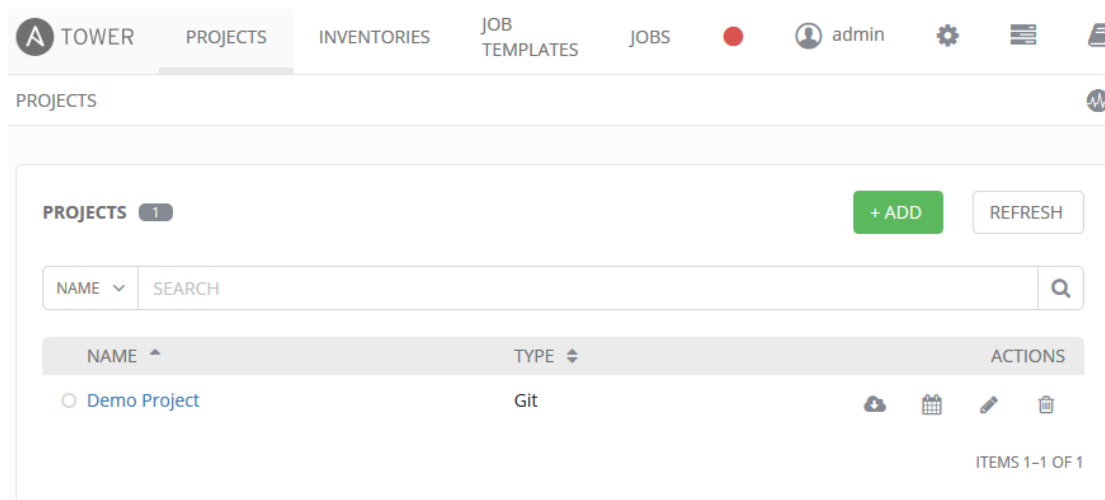
7.7 Project 项目管理

项目是 Ansible 剧本在 Tower 中的一个逻辑的集合。可以通过手动将 Playbook 放到 Tower 服务器项目的默认基础路径或者通过将 Playbook 放到源代码管理系统 (Git, SVN 和 Mercurial) 的方式来管理 Playbook 和 Playbook 目录。

注: 项目的默认基础路径是 `/var/lib/awx/projects`, Tower 管理员可以通过修改配置文件 `/etc/tower/settings.py` 来更改合适的路径, 如下:

```
# vim /etc/tower/settings.py
.....
PROJECTS_ROOT = '/var/lib/awx/projects'    #修改此行等号后面的路径
.....
```

安装完 Ansible Tower 后, 会有一个默认的项目, 点击左上角的“PROJECTS”可以看到, 如下:



点击“Demo Project”, 查看项目的详细信息, 如下:

DEMO PROJECT

DETAILS PERMISSIONS NOTIFICATIONS

* NAME: Demo Project DESCRIPTION: ORGANIZATION: Default

* SCM TYPE: Git

SOURCE DETAILS

* SCM URL: https://github.com/ansible/ansit SCM BRANCH: SCM CREDENTIAL:

SCM UPDATE OPTIONS

- ☐ Clean
- ☐ Delete on Update
- ☒ Update on Launch

CACHE TIMEOUT (SECONDS): 0

CANCEL SAVE

从上面的图片可以看到项目的信息如下：

- 项目名称：Demo Project
- 组织：Default
- 源代码管理方式：Git 版本控制
- 源代码管理系统 URL：https://github.com/ansible/ansible-tower-samples

7.7.1 创建新项目

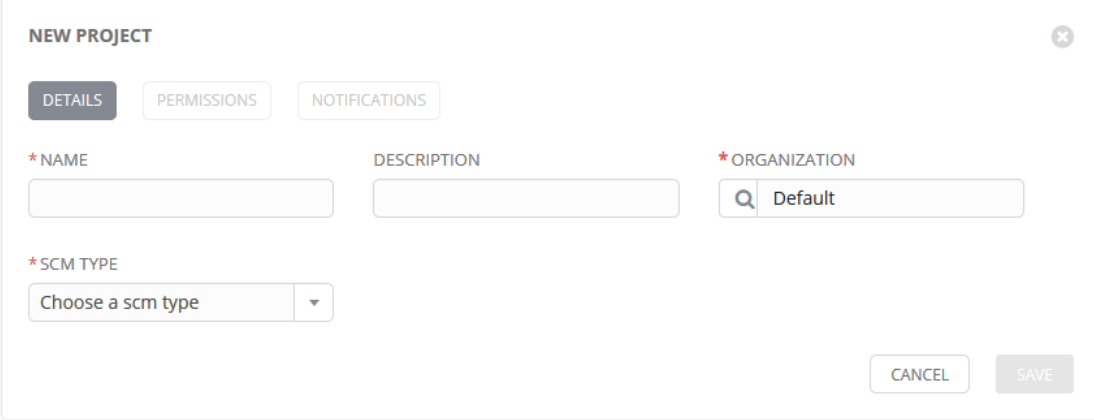
在创建项目的时候需要指定源代码管理方式，如果使用手动管理，则需要将 Playbook 拷贝到 Tower 服务器项目的默认基础路径（默认是/var/lib/awx/projects），如果使用版本控制系统，则需要将 Playbook 上传至版本控制系统。

以下示例为一个简单的手动管理的 Playbook：

```
# pwd
/var/lib/awx/projects
# ls
hello_world
# tree -C hello_world
hello_world
├── site.yml
```


0 directories, 1 file

点击左上角的“PROJECTS”→“+ADD”，创建新的项目，如下：



NEW PROJECT

DETAILS PERMISSIONS NOTIFICATIONS

*NAME DESCRIPTION *ORGANIZATION

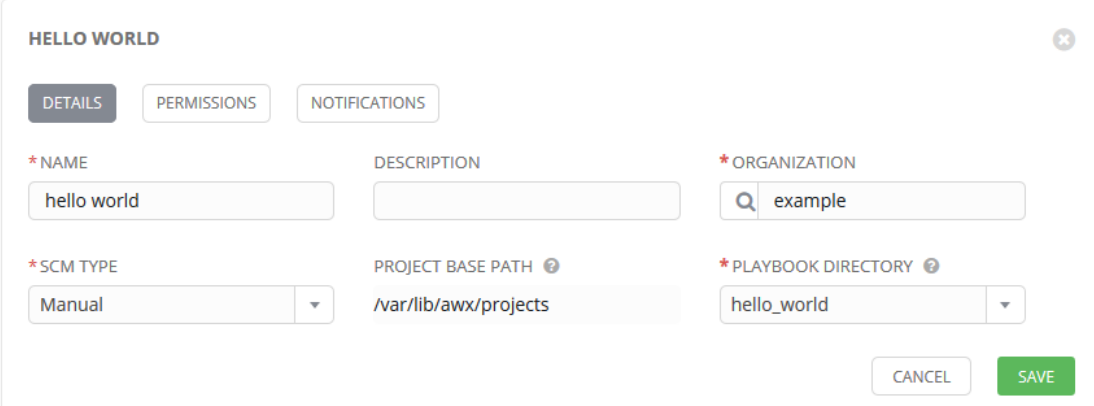
Choose a scm type

CANCEL SAVE

输入项目的信息，*为必填内容，如下：

- NAME：项目名称
- DESCRIPTION：描述
- ORGANIZATION：组织
- SCM TYPE：源代码管理方式
- PROJECT BASE PATH：项目基础路径
- PLAYBOOK DIRECTORY：Playbook 目录

注：一个 Playbook 目录只能属于一个项目，所以如果下拉按钮看不到 Playbook 目录，则可能是默认基础项目路径下没有 Playbook 目录，或者所有的 Playbook 目录都已经分配给了其他项目。



HELLO WORLD

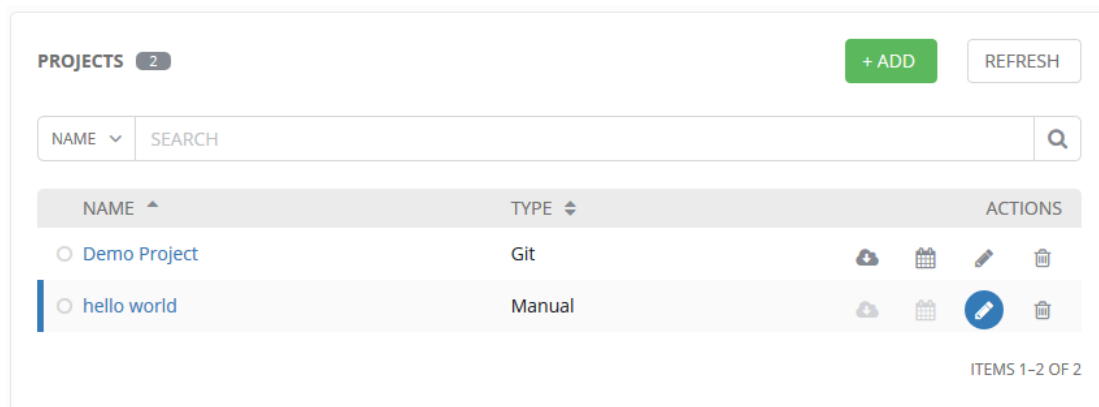
DETAILS PERMISSIONS NOTIFICATIONS

*NAME DESCRIPTION *ORGANIZATION

*SCM TYPE PROJECT BASE PATH ? *PLAYBOOK DIRECTORY ?

CANCEL SAVE

点击“SAVE”保存，可以看到下面多了一个项目，如下：



7.7.2 Playbook 剧本的管理方式

在 Tower 中 Playbook 的管理方式有两种：手动管理和版本控制系统管理。

7.7.2.1 手动管理

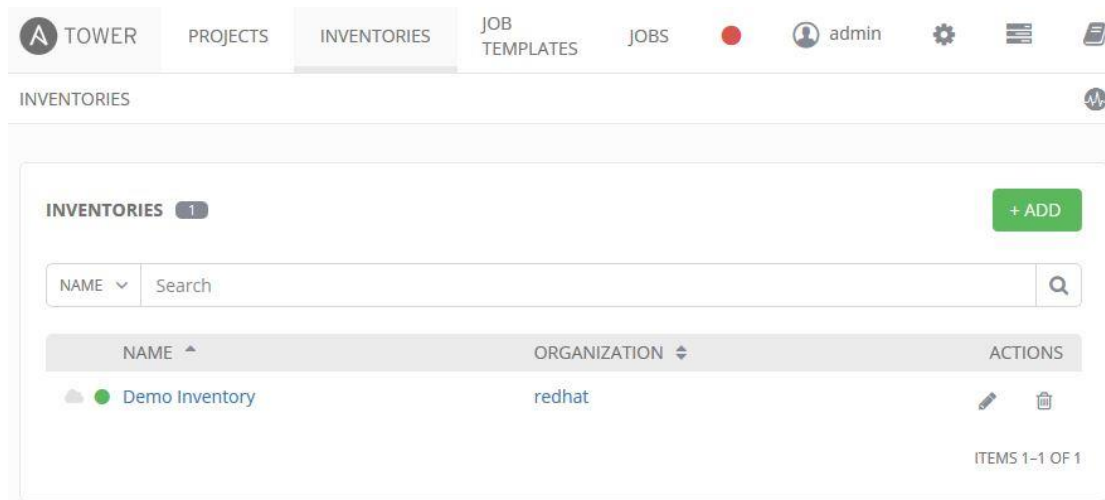
- 1) 在项目基础路径（默认是/var/lib/awx/projects）下创建一个或者多个目录用于存储 Playbooks
- 2) 在项目路径下 Playbook 目录下创建或者复制已有的 Playbook 文件到项目路径下 Playbook 目录中
- 3) 确保项目路径下 Playbook 目录和文件拥有相同的用户和组权限
- 4) 确保项目路径下 Playbook 目录和文件拥有正确的权限

7.7.2.2 SCM 源代码控制管理

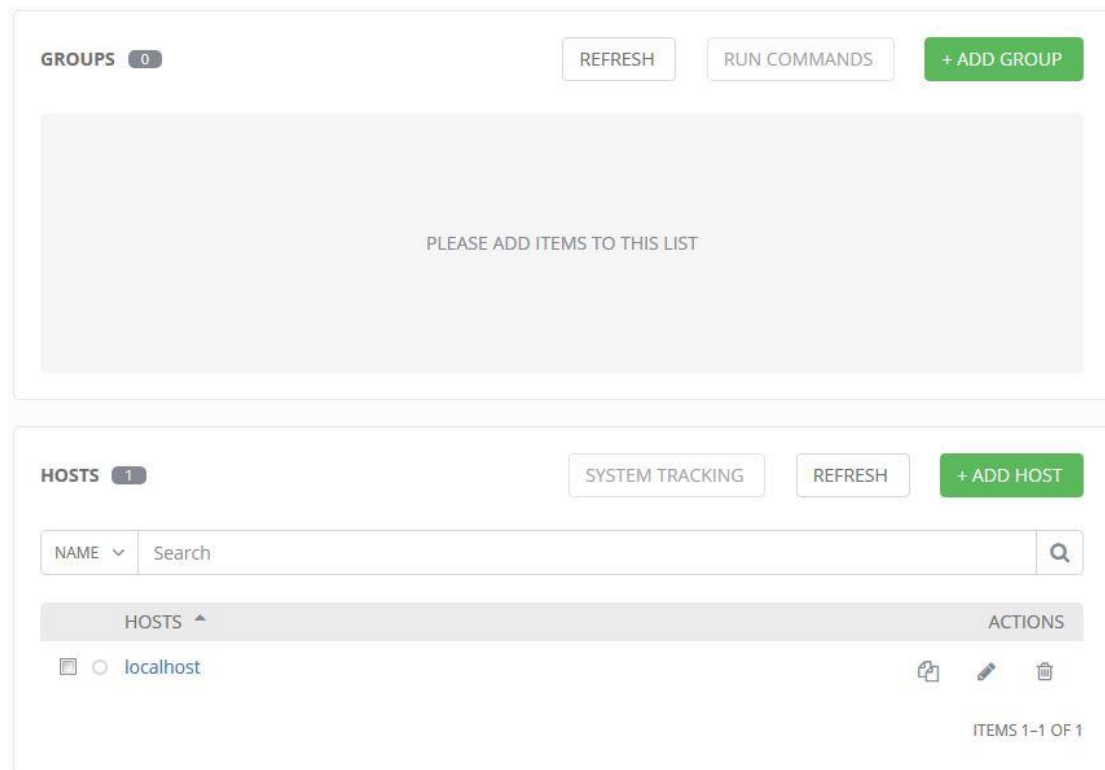
- 1) SCM 路径
- 2) SCM 分支（使用 Git 或 Mercurial 时可选）
- 3) 修订记录（使用 SVN 时可选）
- 4) SCM 凭证（如果需要身份验证，选择适当的 SCM 凭证）
- 5) SCM 更新选项：
 - Clean: 执行更新之前移除本地的修改
 - Delete on Update: 执行更新之前删除本地仓库
 - Update on Launch: 当一个任务使用这个项目，在运行这个任务之前执行更新到本地仓库

7.8 Inventory 清单管理

Ansible 可以同时操作属于一个组的多台主机，组和主机之间的关系通过 Inventory 文件配置。安装完 Ansible Tower 后，会有一个默认的 Inventory，点击主页左上方的“INVENTORIES”，如下：



点击默认的 Inventory: Demo Inventory，可以看到有一台名为 localhost 的 HOST，如下：



点击“localhost”，查看详细信息，如下：

LOCALHOST ☒ ON

*HOST NAME ? DESCRIPTION

localhost

VARIABLES ? ☒ YAML ☐ JSON

```
1 ansible_connection: local
```

CANCEL SAVE

注：HOSTNAME 可以是主机名，IP 地址，或者主机名:端口，IP 地址:端口，如果使用主机名，则需要 ansible tower 主机能够解析到该主机名。

7.8.1 创建 Inventory

点击主页左上方的“INVENTORIES”→“+ADD”，如下：

NEW INVENTORY

DETAILS PERMISSIONS

*NAME DESCRIPTION *ORGANIZATION

SOE SOE Inventory example

VARIABLES ? ☒ YAML ☐ JSON

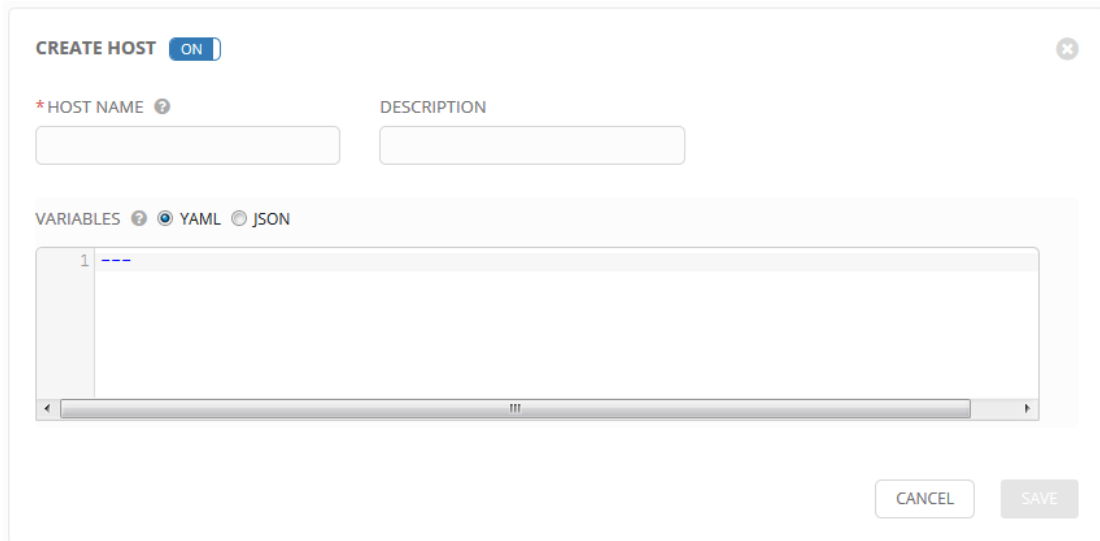
```
1 ---
```

CANCEL SAVE

输入 Inventory 信息，ORGANIZATION 可以点击放大镜进行选择，点击“SAVE”保存。

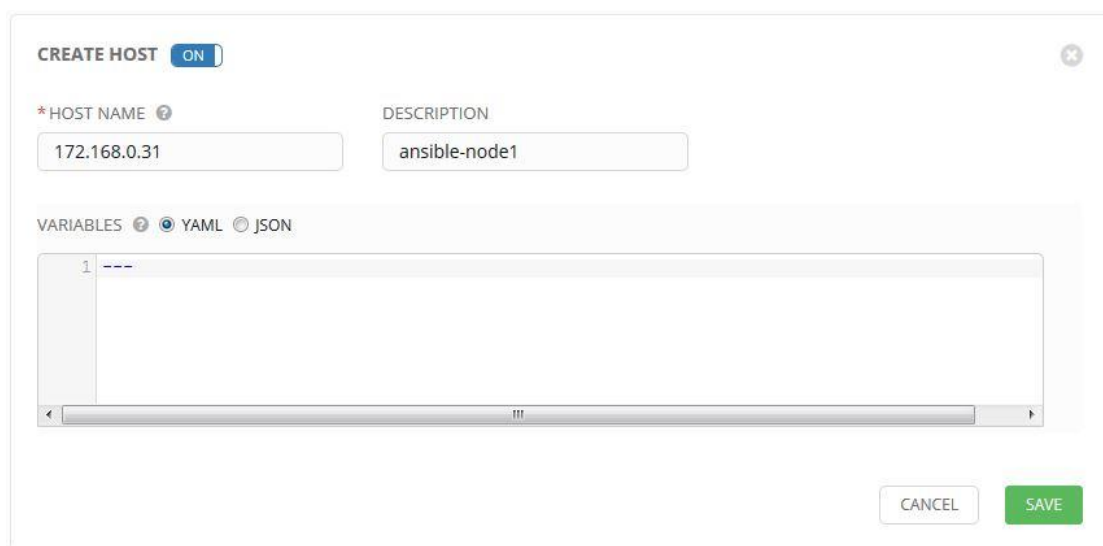
7.8.2 在 Inventory 中添加 Host

在创建新的 Inventory 后会直接进入添加 HOST 和 GROUP 的页面，点击主页左上方的“INVENTORIES”，点击需要添加 HOST 的 Inventory 名称进入，点击“HOSTS”后面的“+ADD HOST”，添加新的 HOST。如下：

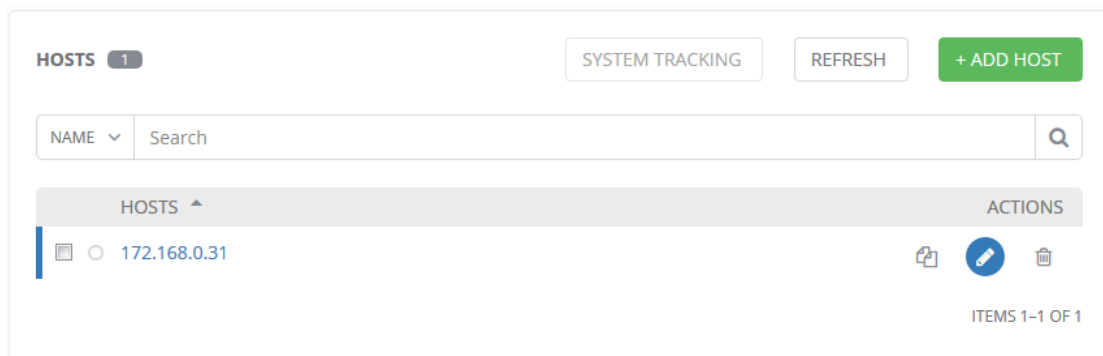


输入 HOST 信息，变量可以根据需求来添加，如下：

- HOSTNAME：主机名或者 IP
- DESCRIPTION：描述信息
- VARIABLES：变量，可根据实际需求填写，例如：HOSTNAME 写的是主机名，但是 Tower 服务器解析不到该主机名，可以使用内置变量 `ansible_ssh_host` 指定 IP



点击“SAVE”保存，如下：



7.8.3 在 Inventory 中添加 Group

在创建新的 Inventory 后会直接进入添加 HOST 和 GROUP 的页面，点击主页左上方的“INVENTORIES”，点击需要添加 GROUP 的 Inventory 名称进入，点击“GROUPS”后面的“+ADD GROUP”，添加新的 GROUP，如下：

输入 GROUP 信息，变量可以根据需求来添加，如下：

- NAME: 组名
- DESCRIPTION: 描述信息
- SOURCE: 组中的 Host 可以手动添加，也可以从其他云平台同步过来
- VARIABLES: 变量，可根据实际需求填写，会对组中的所有 HOST 有效

CREATE GROUP

DETAILS NOTIFICATIONS

*NAME DESCRIPTION SOURCE

soe-test soe test group Manual

VARIABLES ? ? ☒ YAML ☐ JSON

1

CANCEL SAVE

点击“SAVE”保存，如下：

GROUPS 1 REFRESH RUN COMMANDS + ADD GROUP

NAME Search

GROUPS ACTIONS

soe-test

ITEMS 1-1 OF 1

7.8.4 在 Group 中添加子 Group 和 Host

GROUP 中可以添加 HOST，也可以添加子 GROUP，在子 GROUP 中可以继续添加子 GROUP，支持多层嵌套。

注：在一个 Inventory 中，可以包含多个 GROUP 和子 GROUP，但是多层次的 GROUP 和子 GROUP 之间不能有重名的 GROUP，不同的 Inventory 之间可以有重名的 GROUP。

为 GROUP 添加子 GROUP 和 HOST，可以在 GROUP 中添加新的子 GROUP 和 HOST，也可以将已经存在的子 GROUP 和 HOST 复制或者移动到 GROUP。

注：如果将一个子 GROUP 移动或复制到另一个 GROUP，那么这个子 GROUP 中包含的子 GROUP 和 HOST 会继承移动或复制操作。

7.8.4.1 添加新的子 GROUP 和 HOST 到 GROUP

在 GROUP 中添加子 GROUP 与在 Inventory 中添加 GROUP 的方法一样，点击需要添加子 GROUP 的 GROUP 名称进入，点击 GROUPS 后面的“+ADD GROUP”添加即可。

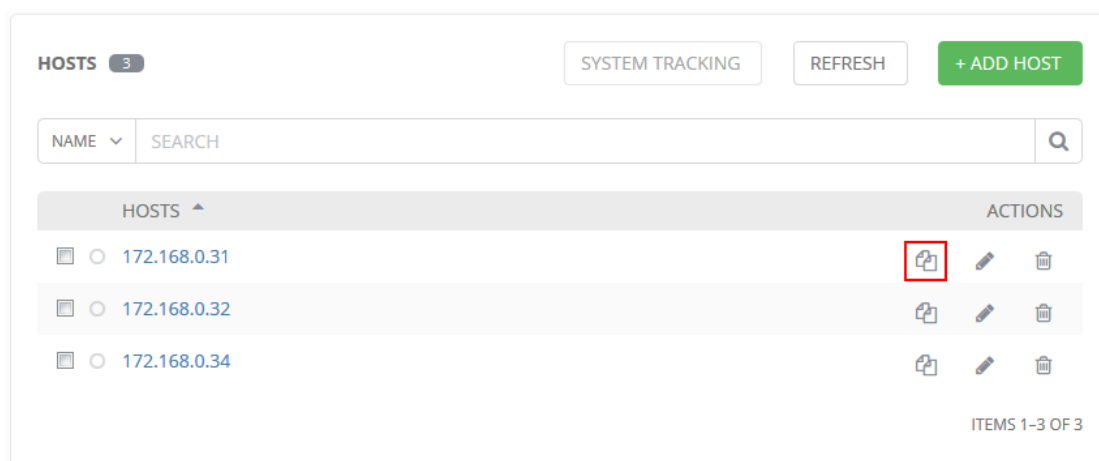
在 GROUP 中添加 HOST 与在 Inventory 中添加 HOST 的方法一样，点击需要添加 HOST 的 GROUP 名称进入，点击 HOSTS 后面的“+ADD HOST”添加即可。

7.8.4.2 移动或复制 GROUP 和 HOST 到 GROUP

移动或者复制 GROUP 和 HOST 只能在 Inventory 内操作，不可跨 Inventory 操作。

1) 复制 HOST

复制 GROUP 中的 HOST 不需要进入 GROUP 中，直接在 Inventory 中的 HOST 中就可以操作，点击需要操作的 Inventory 进入，在 HOSTS 下点击需要复制的 HOST 后面的复制移动操作按钮，例如，复制 172.168.0.31，如下：



选择“Copy”，选择需要复制到的 GROUP，例如，复制到 test，如下：

172.168.0.31

☒ Copy ☐ Move

TARGET GROUP NAME SEARCH

TARGET GROUP NAME ^

- ☐ rhel6
- ☐ rhel7
- ☐ soe
- ☐ soe-test
- ☒ test

ITEMS 1-5 OF 5

☐ Use the inventory root

CANCEL SAVE

点击“SAVE”保存即可，进入 test 组就可以看到刚刚复制的 HOST。

2) 移动 HOST

如果要将一个 GROUP 中的 HOST 移动到另一个 GROUP 中，则需要进入 HOST 所在的 GROUP 中进行操作，进入 Inventory “SOE” 的 GROUP “test” 中，如下：

INVENTORIES / SOE / TEST

GROUPS 0 REFRESH RUN COMMANDS + ADD GROUP

PLEASE ADD ITEMS TO THIS LIST

HOSTS 2 SYSTEM TRACKING REFRESH + ADD HOST

NAME SEARCH

HOSTS ^ ACTIONS

<input type="checkbox"/>	<input type="radio"/>	172.168.0.32	<input type="image"/>	<input type="image"/>	<input type="image"/>
<input type="checkbox"/>	<input type="radio"/>	172.168.0.34	<input type="image"/>	<input type="image"/>	<input type="image"/>

ITEMS 1-2 OF 2

点击需要移动的 HOST 后面的复制移动操作按钮，例如，移动 172.168.0.34，如下：

HOSTS 2 SYSTEM TRACKING REFRESH + ADD HOST

NAME SEARCH

HOSTS ^ ACTIONS

<input type="checkbox"/>	<input type="radio"/>	172.168.0.32	<input type="image"/>	<input type="image"/>	<input type="image"/>
<input type="checkbox"/>	<input type="radio"/>	172.168.0.34	<input checked="" type="image"/>	<input type="image"/>	<input type="image"/>

ITEMS 1-2 OF 2

选择“MOVE”，选择需要移动到的 GROUP，例如，rhel6，如下：

172.168.0.34

Copy

Move

TARGET GROUP NAME

SEARCH

TARGET GROUP NAME ^

rhel6

rhel7

soe

soe-test

test

ITEMS 1-5 OF 5

☐ Use the inventory root

CANCEL

SAVE

点击“SAVE”保存，HOST“172.168.0.34”就被移动到 rhel6 中了，如果原来的 GROUP 中仍然可以看到 HOST“172.168.0.34”，说明 HOST“172.168.0.34”属于原 GROUP 中的某个子 GROUP 中。

3) 复制 GROUP

复制一个 GROUP 到另一个 GROUP 中，那么这个 GROUP 中的子 GROUP 和 HOST 同样也会继承复制过去。如果要复制一个 GROUP，需要进入这个 GROUP 的上一级 GROUP，如果上一级没有 GROUP，那么就进入该 GROUP 所在的 Inventory 根下操作，例如，复制 Inventory“SOE”下 GROUP“soe-test”下的子 GROUP“rhel6”，进入 Inventory“SOE”下 GROUP“soe-test”中，如下：

[INVENTORIES](#) / [SOE](#) / [SOE-TEST](#)

GROUPS 2

REFRESH

RUN COMMANDS

+ ADD GROUP

NAME ^

SEARCH

GROUPS ^

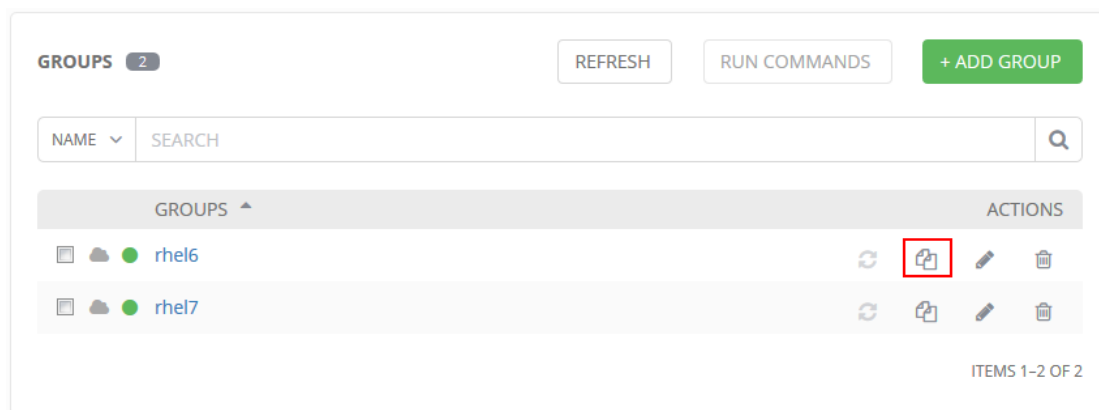
ACTIONS

rhel6

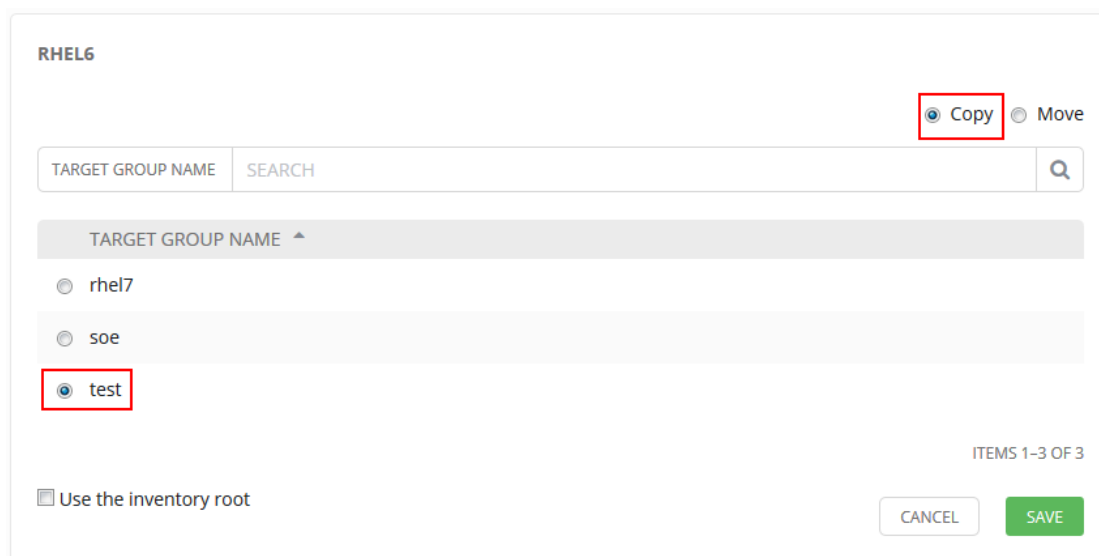
rhel7

ITEMS 1-2 OF 2

点击 GROUP“rhel6”后面的复制移动操作按钮，如下：



选择“Copy”，选择需要复制到的 GROUP，例如，test，如下：



点击“SAVE”保存即可，GROUP“rhel6”就被复制到 GROUP“test”里了，包括 GROUP“rhel6”中的 HOST。

4) 移动 GROUP

移动一个 GROUP 到另一个 GROUP 中，那么这个 GROUP 中的子 GROUP 和 HOST 同样也会继承移动过去。如果要移动一个 GROUP，也需要进入这个 GROUP 的上一级 GROUP，如果上一级没有 GROUP，那么就进入该 GROUP 所在的 Inventory 根下操作，例如，移动 Inventory“SOE”下 GROUP“soe-test”下的子 GROUP“rhel7”，进入 Inventory“SOE”下 GROUP“soe-test”中，如下：

GROUPS 2

REFRESH















RUN COMMANDS

+ ADD GROUP

NAME SEARCH

GROUPS

ACTIONS

   rhel6	   
   rhel7	   

ITEMS 1-2 OF 2

点击 GROUP “rhel7” 后面的复制移动操作按钮，如下：

GROUPS 2

REFRESH















RUN COMMANDS

+ ADD GROUP

NAME SEARCH

GROUPS

ACTIONS

   rhel6	   
   rhel7	   

ITEMS 1-2 OF 2

选择 “Move”，选择需要移动到的 GROUP，例如，test，如下：

RHEL7

☐ Copy ☒ Move

TARGET GROUP NAME SEARCH

TARGET GROUP NAME

☐ rhel6

☐ soe

☒ test

ITEMS 1-3 OF 3

☐ Use the inventory root

CANCEL

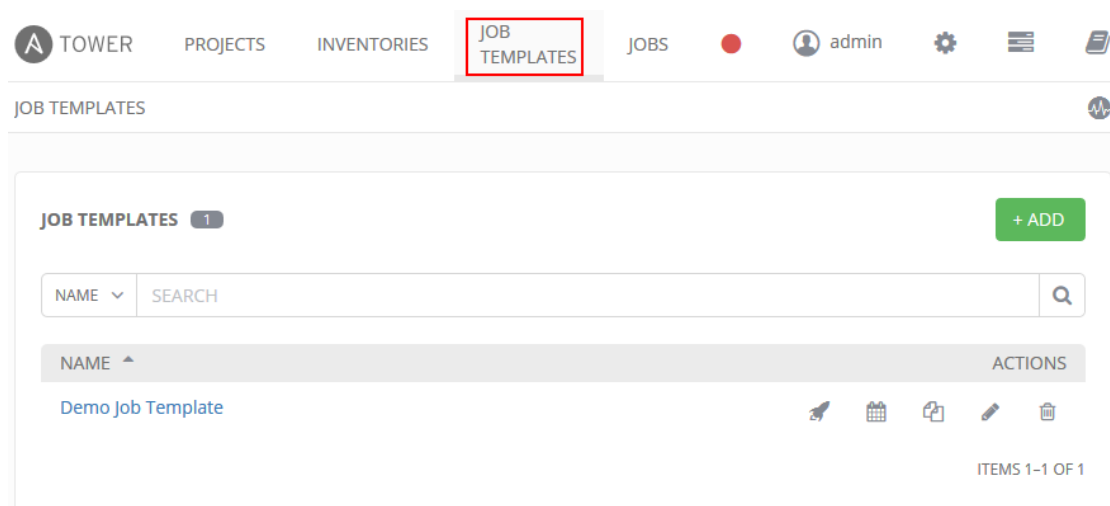
SAVE

点击 “SAVE” 保存即可，GROUP “rhel7” 就被移动到 GROUP “test” 中了，包括 GROUP “rhel7” 中的 HOST。

7.9 Job Template 作业模板管理

作业模板是用来定义运行 Ansible 作业的设置和参数，作业模板用来在不同的时间执行相同的作业，同时也鼓励 Ansible Playbook 的重用。

在安装完 Ansible Tower 后，会有一个默认的作业模板，点击页面上方的“JOB TEMPLATES”可以看到，如下：



7.9.1 创建新作业模板

点击“JOB TEMPLATES” → “+ADD”，添加一个新的作业模板，如下：

- Check: 不执行 Playbook, 会检查 Playbook 语法, 以及会在指定的主机上做出的改变
- Scan: 在指定主机上执行 Playbook, 且会存储扫描的信息用于 Tower 的系统跟踪特性
- INVENTORY: 清单, 用于指定执行 Playbook 的主机列表
- PROJECT: 项目, 用于指定作业模板使用哪个项目中的 Playbook
- PLAYBOOK: Playbook, 指定作业模板使用项目中的哪个 Playbook
- MACHINE CREDENTIAL: Machine 凭证, 登录远程主机使用的凭证
- CLOUD CREDENTIAL: Cloud 凭证, 传递给正在运行的 Playbook
- NETWORK CREDENTIAL: Network 凭证, 登录和管理网络设备使用的凭证
- FORKS: 并发数, 同时执行 Playbook 的主机数量, 0 为表示使用 Ansible 默认值, Ansible 默认值为 5
- LIMIT: 对清单中的主机进行限制, 多个用冒号隔开, a:b 表示 a 或者 b, a:b&c 表示 a 或者 b 但是必须 c, a:!b 表示 a 但是没有 c
- VERBOSITY: 信息显示, 用于控制 Playbook 的输出内容, 0 为默认值, 表示正常输出, 数字越大, 输出内容越多, 越详细, 主要用于 Debug 分析。
- JOB TAGS: Playbook 的 Tag, 指定 Playbook 中需要执行的 Tags, 多个用逗号隔开, 在只需要执行 Playbook 中的某一些任务时使用
- SKIP TAGS: Playbook 的 Tag, 指定 Playbook 中不需要执行的 Tags, 多个用逗号隔开, 在需要跳过 Playbook 中的某一些任务时使用
- OPTIONS: 选项
 - Enable Privilegr Escalation: 开启特权升级
 - Allow Provisioning Callbacks: 开启配置回调
- LABELS: 标签, 用于对任务模板进行标识
- EXTRA VARIABLES: 额外变量, 传递额外的变量时使用, 且拥有最高的优先级

NEW JOB TEMPLATE

DETAILS

COMPLETED JOBS

PERMISSIONS

NOTIFICATIONS

*NAME

hello world

DESCRIPTION

hello world

*JOB TYPE ?

Run

☐ Prompt on launch

*INVENTORY ?

Q

SOE

☐ Prompt on launch

*PROJECT ?

Q

hello world

☐ Prompt on launch

*PLAYBOOK ?

site.yml

☐ Prompt on launch

*MACHINE CREDENTIAL ?

Q

SSH

☐ Prompt on launch

CLOUD CREDENTIAL ?

Q

☐ Prompt on launch

NETWORK CREDENTIAL ?

Q

☐ Prompt on launch

FORKS ?

0

☐ Prompt on launch

LIMIT ?

☐ Prompt on launch

*VERBOSITY ?

0 (Normal)

☐ Prompt on launch

JOB TAGS ?

☐ Prompt on launch

SKIP TAGS ?

☐ Prompt on launch

OPTIONS

☐ Enable Privilege Escalation ?

☐ Allow Provisioning Callbacks ?

LABELS ?

EXTRA VARIABLES ? ? ?

YAML

JSON

1

☐ Prompt on launch

ADD SURVEY

CANCEL

SAVE

点击“SAVE”保存，就可以看到多出一个任务模板，如下：



从 Ansible Tower 3.0 开始，任务模板支持复制，如下：



但是复制功能并不能复制已经分配的调度，通知以及权限，这些需要用户重启配置。

7.9.2 运行作业

Ansible Tower 的好处就是使用按钮即可部署一个 Ansible Playbook，点击页面上方的“JOB TEMPLATES”，准备运行一个作业，例如，“hello world”，如下：



点击“hello world”后面的第一个按钮，开始运行，如下：

RESULTS

STATUS

☐

Pending

TEMPLATE

hello world

JOB TYPE

Run

LAUNCHED BY

admin

INVENTORY

SOE

PROJECT

hello world

PLAYBOOK

site.yml

MACHINE CREDENTIAL

SSH

VERBOSITY

Default

EXTRA VARIABLES

STANDARD OUT

Waiting for results...

DETAILS

1

Please select from a play below to view its associated tasks.

PLAY NAME

Q

ALL

FAILED

Waiting...

2

Please select a task below to view its associated hosts

TASK NAME

Q

ALL

FAILED

Waiting...

3

Please select a host below to view associated task details.

HOST NAME

Q

ALL

FAILED

Waiting...

EVENT SUMMARY

等待一段时间，刷新页面，查看是否运行成功，不同 Playbook 运行所花费的时间是不一样的，取决于 Playbook 中任务模块的多少，以及在远程主机上执行的任务类型等。

作业运行完成后的界面如下：

RESULTS

STATUS

2017-02-06 晚上8点56分11秒

2017-02-06 晚上8点56分22秒

00:00:11

hello world

SSH

TEMPLATE

JOB TYPE

LAUNCHED BY

INVENTORY

PLAYBOOK

VERBOSITY

hello world

Run

admin

SOE

site.yml

Default

EXTRA VARIABLES

1

DETAILS

1

Please select from a play below to view its associated tasks.

PLAY NAME

Q

ALL

FAILED

PLAYS

STARTED

ELAPSED

all

20:56:17

00:00:04

2

Please select a task below to view its associated hosts

TASK NAME

Q

ALL

FAILED

TASKS

STARTED

ELAPSED

HOST STATUS

setup

20:56:17

00:00:02

3

Hello World!

20:56:20

00:00:01

3

3

Please select a host below to view associated task details.

HOST NAME

Q

ALL

FAILED

HOSTS

ITEM

MESSAGE

172.168.0.31

172.168.0.32

172.168.0.34

EVENT SUMMARY

4

Please select a host below to view a summary of all associated tasks.

HOST NAME

Q

ALL

FAILED

HOSTS

COMPLETED TASKS

172.168.0.31

2

172.168.0.32

2

172.168.0.34

2

HOST STATUS SUMMARY

OK: 100%

STANDARD OUT

PLAY [all]

TASK [setup]

TASK [Hello World!]

PLAY RECAP

ok: [172.168.0.31]

ok: [172.168.0.34]

ok: [172.168.0.32]

ok: [172.168.0.31] => {
 "msg": "Hello World!"
}
ok: [172.168.0.32] => {
 "msg": "Hello World!"
}
ok: [172.168.0.34] => {
 "msg": "Hello World!"
}
ok: [172.168.0.31] : ok=2 changed=0 unreachable=0 failed=0
172.168.0.32 : ok=2 changed=0 unreachable=0 failed=0
172.168.0.34 : ok=2 changed=0 unreachable=0 failed=0

7.9.3 计划作业

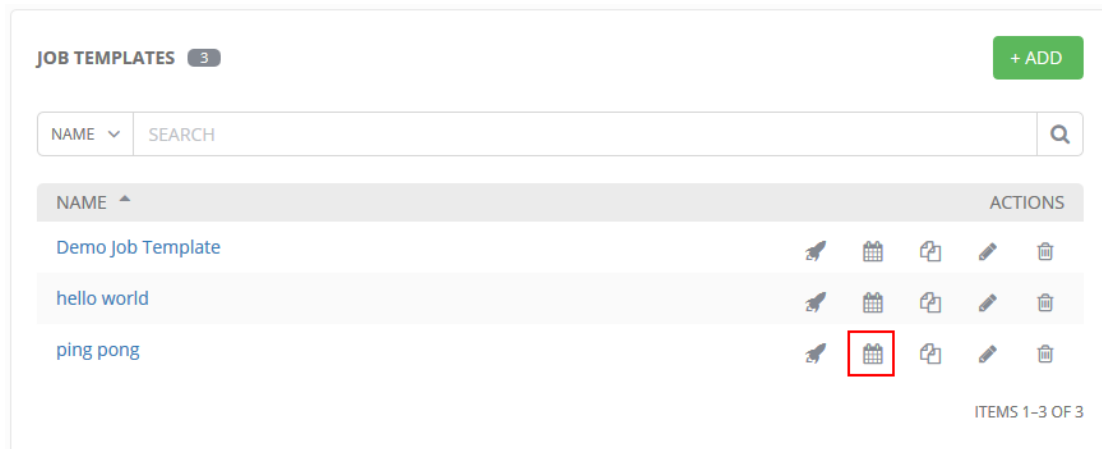
作业模板可以配置为在预定的计划时间点自动运行。

7.9.3.1 创建一个计划作业

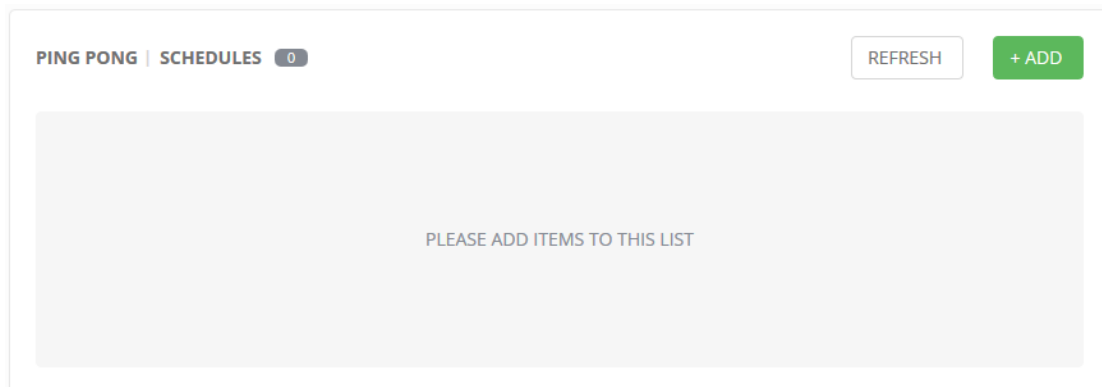
先创建一个新的项目“ping pong”，再创建一个新的作业模板“ping pong”使用项目“ping pong”中的 Playbook，点击上方的“JOB TEMPLATES”，如下：



点击“ping pong”后面的第二个按钮配置计划作业，如下：



现在“ping pong”还没有配置过任何调度计划。



点击“+ADD”，创建一个调度计划，如下：

ADD SCHEDULE

* NAME: Schedule name
A schedule name is required.

* START DATE: 2017-2-06

* START TIME (HH24:MM:SS): 00:00:00

* LOCAL TIME ZONE: Asia/Shanghai

* REPEAT FREQUENCY: None (run once)

The scheduler options are invalid or incomplete.

EXTRA VARIABLES: ☒ YAML ☐ JSON

1 ---

CANCEL SAVE

输入调度计划的信息，带*为必填内容，如下：

- NAME：调度计划的名称
- START DATE：计划开始的日期
- START TIME (HH24:MM:SS)：计划开始的时间（24 小时格式）
- LOCAL TIME ZONE：本地时区
- REPEAT FREQUENCY：重复频率
 - None (run once)：不重复，只运行一次
 - Minute：按分钟重复，可以配置每几分钟重复一次，是否结束，多少次之后结束，指定日期结束
 - Hour：按小时重复，可以配置每几小时重复一次，是否结束，多少次之后结束，指定日期结束
 - Day：按天重复，可以配置每几天重复一次，是否结束，多少次之后结束，指定日期结束
 - Week：按周重复，可以配置每周的哪几天重复，是否结束，多少次之后结束，指定日期结束
 - Month：按月重复，可以配置每月的哪天重复一次，每月的第几个星期几重复一次，是否结束，多少次之后结束，指定日期结束

- Year 按年重复，可以配置每年的第几个月的第几个天重复一次，每年的第几个月的第几个星期几重复一次，是否结束，多少次之后结束，指定日期结束
- EXTAR VARIABLES: 额外变量

PING

* NAME

ping

* START DATE

2017-2-06

* START TIME (HH24:MM:SS)

21

:

45

:

00

* LOCAL TIME ZONE

Asia/Shanghai

* REPEAT FREQUENCY

Day

FREQUENCY DETAILS

* EVERY

1

DAYS

* END

Never

SCHEDULE DESCRIPTION

every day

OCCURRENCES (Limited to first 10)

DATE FORMAT ☒ LOCAL TIME ☐ UTC

2017-02-06 21:45:00 CST
2017-02-07 21:45:00 CST
2017-02-08 21:45:00 CST
2017-02-09 21:45:00 CST
2017-02-10 21:45:00 CST
2017-02-11 21:45:00 CST
2017-02-12 21:45:00 CST
2017-02-13 21:45:00 CST
2017-02-14 21:45:00 CST
2017-02-15 21:45:00 CST

EXTRA VARIABLES ☒ YAML ☐ JSON

1

CANCEL

SAVE

以上示例配置了“ping pong”从 2017-02-06 21:45:00 开始，每天执行一次，用不停止，并且显示了最近 10 次运行的时间。

点击“SAVE”保存即可。

PING PONG | SCHEDULES 1
REFRESH
+ ADD

NAME	SEARCH	
NAME	NEXT RUN	FINAL RUN ACTIONS
ON ping	2017-02-06 晚上9点45分0秒	

ITEMS 1-1 OF 1

保存后，可以看到“ping pong”的调度计划，并且显示了下一次运行的时间为 2017-02-06 晚上 9 点 45 分 0 秒。

7.10 Job 作业管理

Tower 在 Inventory 中的主机上运行一个 Ansible Playbook 被称作为作业。

点击页面上方的“JOBS”，可以看到已经运行过，和正在运行的作业的状态，如下：

TOWER
PROJECTS
INVENTORIES
JOB TEMPLATES
JOBS
admin

JOBS

JOBS		SCHEDULES		REFRESH
NAME	SEARCH			
ID	NAME	FINISHED	LABELS	ACTIONS
4	ping pong	2017-02-06 晚上9点45分38秒		
3	hello world	2017-02-06 晚上8点56分22秒		
2	Cleanup Activity Stream	2017-01-03 下午4点34分25秒		
1	Cleanup Activity Stream	2016-12-20 下午4点34分53秒		

ITEMS 1-4 OF 4

点击“SCHEDULES”可以看到已经运行过，和正在运行的计划作业的状态，如下：

TOWER

PROJECTS

INVENTORIES

JOB TEMPLATES

JOB

admin

JOB

JOB

SCHEDULES

REFRESH

NAME

SEARCH

NAME	TYPE	NEXT RUN	ACTIONS
<div>ON</div> Cleanup Activity Stream	Management Job	2017-02-07 下午4点34分19秒	<div></div> <div></div>
<div>ON</div> ping pong	Playbook Run	2017-02-07 晚上9点45分0秒	<div></div> <div></div>
<div>ON</div> Cleanup Job Details	Management Job	2017-02-12 下午4点34分19秒	<div></div> <div></div>
<div>ON</div> Cleanup Fact Details	Management Job	2017-03-01 下午4点34分19秒	<div></div> <div></div>

ITEMS 1-4 OF 4

点击 JOB 的名称，可以查看 JOB 运行的结果。

JOB

SCHEDULES

REFRESH

NAME

SEARCH

ID	NAME	FINISHED	ACTIONS
4	ping pong	2017-02-06 晚上9点45分38秒	<div></div> <div></div>
3	hello world	2017-02-06 晚上8点56分22秒	<div></div> <div></div>
2	Cleanup Activity Stream	2017-01-03 下午4点34分25秒	<div></div>
1	Cleanup Activity Stream	2016-12-20 下午4点34分53秒	<div></div>

ITEMS 1-4 OF 4

例如，查看 JOB “hello world” 的运行结果，点击 “hello world”，如下：

运行结果：

RESULTS

STATUS

2017-02-06 晚上8点56分11秒

2017-02-06 晚上8点56分22秒

00:00:11

hello world

SSH

Successful

2017-02-06 晚上8点56分11秒

2017-02-06 晚上8点56分22秒

00:00:11

hello world

SSH

TEMPLATE

JOB TYPE

LAUNCHED BY

INVENTORY

PLAYBOOK

VERBOSITY

hello world

Run

admin

SOE

site.yml

Default

EXTRA
VARIABLES

1

标准输出：

STANDARD OUT

PLAY [all]

TASK [setup]

TASK [Hello World!]

PLAY RECAP

ok: [172.168.0.31]

ok: [172.168.0.34]

ok: [172.168.0.32]

ok: [172.168.0.31] => {

"msg": "Hello World!"

}

ok: [172.168.0.32] => {

"msg": "Hello World!"

}

ok: [172.168.0.34] => {

"msg": "Hello World!"

}

172.168.0.31 : ok=2 changed=0 unreachable=0 failed=0

172.168.0.32 : ok=2 changed=0 unreachable=0 failed=0

172.168.0.34 : ok=2 changed=0 unreachable=0 failed=0

细节信息：

DETAILS

1

Please select from a play below to view its associated tasks.

PLAY NAME

Q

ALL

FAILED

PLAYS	STARTED	ELAPSED
<div><div></div>all</div>	20:56:17	00:00:04

2

Please select a task below to view its associated hosts

TASK NAME

Q

ALL

FAILED

TASKS	STARTED	ELAPSED	HOST STATUS
<div><div></div>setup</div>	20:56:17	00:00:02	<div>3</div>
<div><div></div>Hello World!</div>	20:56:20	00:00:01	<div>3</div>

3

Please select a host below to view associated task details.

HOST NAME

Q

ALL

FAILED

HOSTS	ITEM	MESSAGE
<div><div></div>172.168.0.31</div>		
<div><div></div>172.168.0.32</div>		
<div><div></div>172.168.0.34</div>		

点击 TASKS，可以查看每个任务的细节信息，例如：

TASKS	STARTED	ELAPSED	HOST STATUS
● setup	20:56:17	00:00:02	3
● Hello World!	20:56:20	00:00:01	3

3 Please select a host below to view associated task details.

HOSTS	ITEM	MESSAGE
● 172.168.0.31		Hello World!
● 172.168.0.32		Hello World!
● 172.168.0.34		Hello World!

点击 HOSTS，可以查看每个主机的细节信息，例如：

HOST EVENT

<div>DETAILS</div> <div>JSON</div>			
EVENT		RESULTS	
HOST	172.168.0.31	MSG	Hello World!
STATUS	● ok	_ANSIBLE_VER	
ID	8	BOSE_ALWAYS	true
CREATED	2017-02-06T12:56:21.534Z	_ANSIBLE_NO	
PLAY	all	_LOG	false
TASK	Hello World!		
MODULE	No result found		
		PREV HOST	NEXT HOST
		CLOSE	

JOB 结果除了可以显示 Playbook 的运行结果，还可以显示云 Inventory 同步的结果和标准输出信息，SCM 源代码管理的同步结果和标准输出信息。

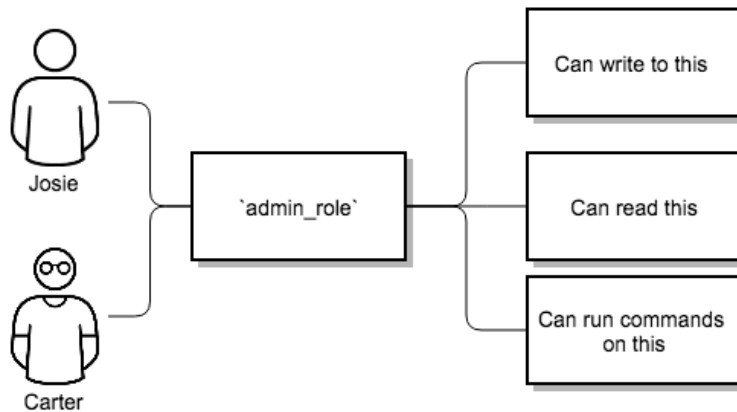
7.11 Permissions 权限管理

Ansible Tower 基于角色的访问控制（RBAC）允许 Tower 管理员访问清单库存，组织，甚至更多，也可以集中管理各种凭证，允许用户利用密钥但又没有将密钥暴露给用户，RBAC 控制允许 Tower 帮助你增加安全性和简化管理。

一个角色实际上是一个功能的集合，用户通过被授予角色或者角色的继承获得允许访问这些功能和 Tower 的资源权限。

7.11.1 角色层次结构和继承

例如：有一个组织叫 “SomeCompany”，想允许两个用户 “Josie” 和 “Carter” 访问管理所有与该组织有关的设置，应该授予这两个用户组织管理角色，如下：

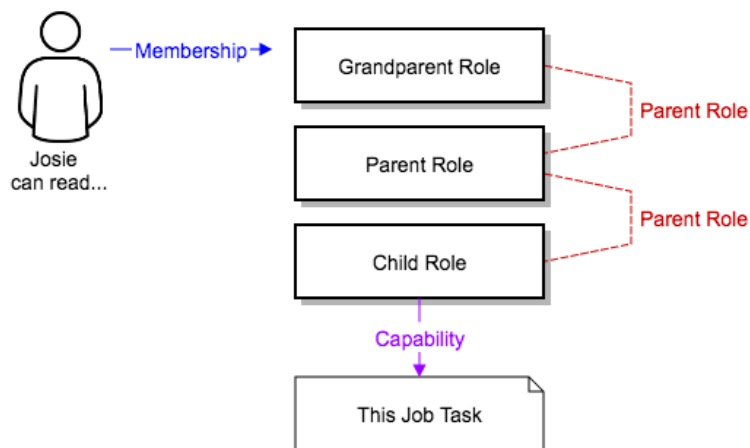


通常，会有许多系统中的角色，有时可能想要一些角色包括所有其他角色的功能。例如：想让一个系统管理员能够访问组织管理员能够访问的一切，以及项目管理员能够访问的一切，等等。

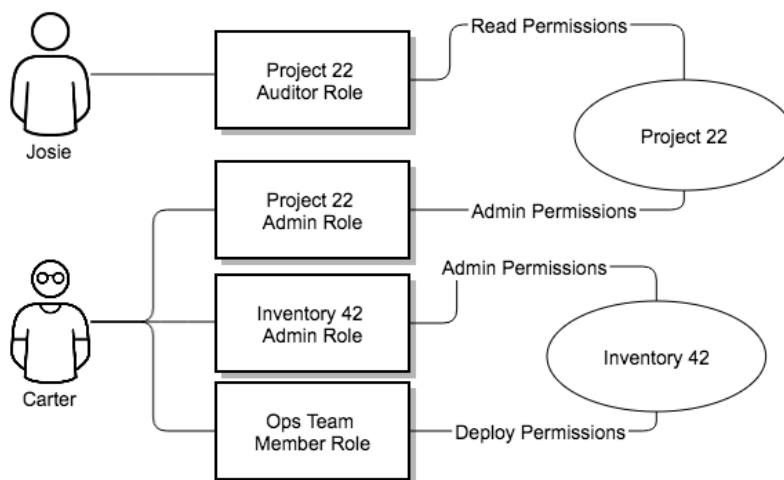
这个概念被称为“角色层次结构”：

- 父角色拥有子角色的所有功能
- 角色中的成员自动拥有角色的功能，以及子角色的功能

角色层次结构允许角色拥有父角色，角色中的任何功能都是隐藏授予父角色的（或者父角色的父角色，等等）。如下：



通常，你会有许多系统中的角色，你希望一些角色包含其他角色的所有功能，例如：你想让一个系统管理员访问所有组织管理员可以访问的，项目管理员可以访问的一切，等等。可以看到，这个涉及到角色层次结构。当然，角色可以拥有一个或者多个父角色，角色中的功能会被隐藏授予所有的父角色。



7.11.2 用户权限

在 Tower 中，用户类型有三种：

- **Normal User:** Normal User 被赋予了角色和权利后，对 inventory 和 projects 具有有限的读写访问权限

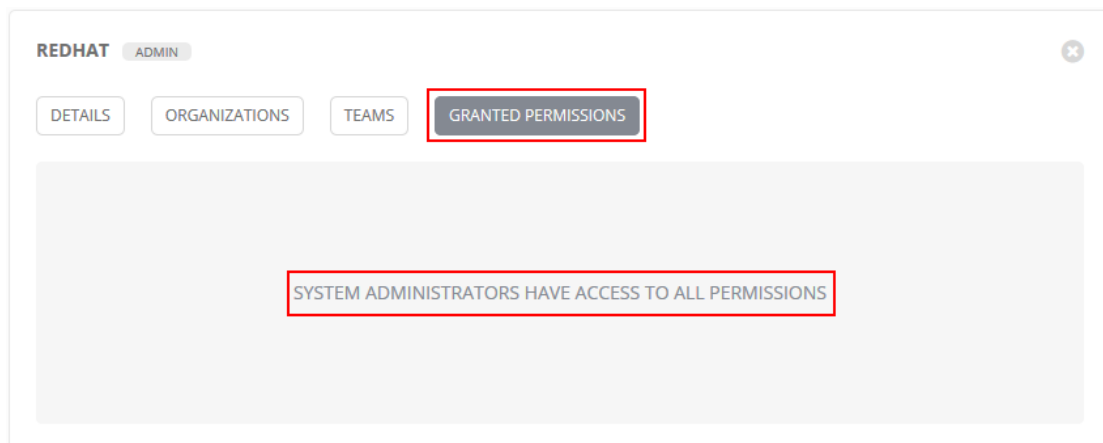
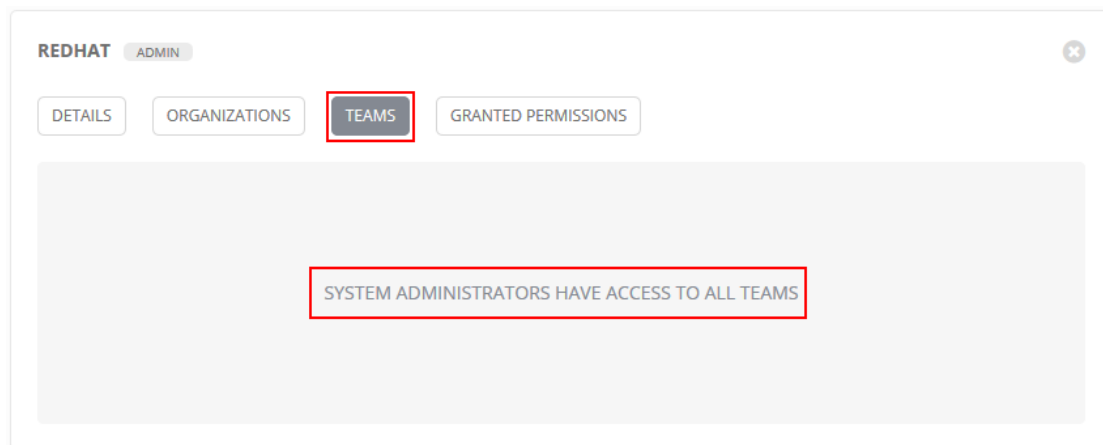
- **System Auditor:** 系统审计员隐藏继承了对整个 Tower 环境中的所有对象的只读权限。
- **System Administrator:** 系统管理员隐藏继承了对整个 Tower 环境中所有对象的读写执行权限。

只有系统管理员可以创建、删除、编辑用户，分配用户类型。在 Tower 环境中可以存在多个系统管理员，被分配了系统管理员的用户，拥有与系统管理员 admin 同等的权利。例如：Tower 中有一个系统用户 redhat，用户类型为系统管理员，用户信息如下：

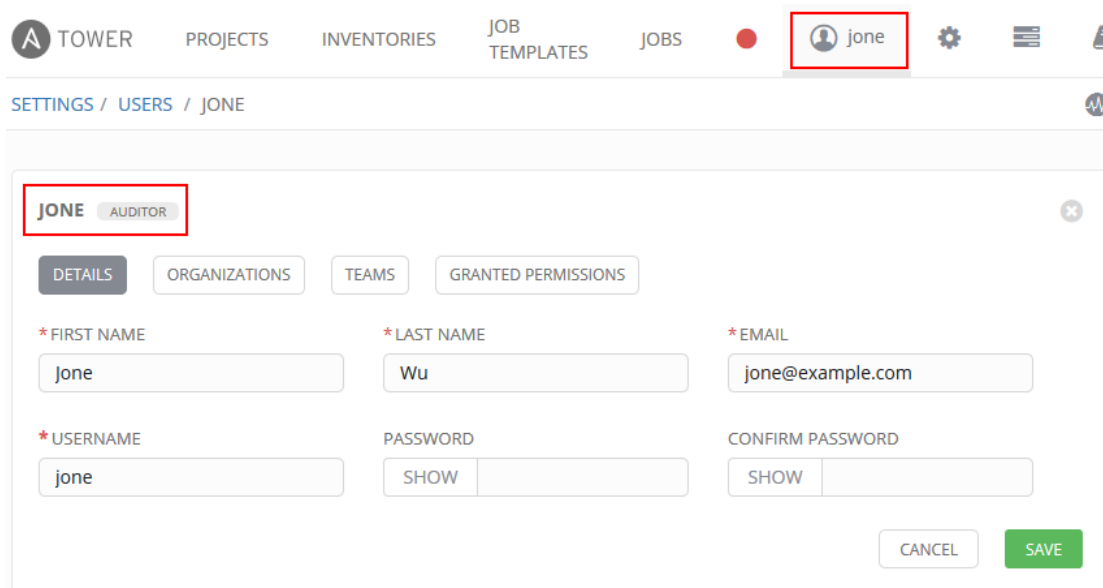
The screenshot shows the 'REDHAT ADMIN' user profile page. It has tabs for 'DETAILS', 'ORGANIZATIONS', 'TEAMS', and 'GRANTED PERMISSIONS'. The 'DETAILS' tab is active. The form includes fields for First Name (Red), Last Name (Hat), Email (redhat@example.com), Username (redhat), Password (with a 'SHOW' toggle), and Confirm Password (with a 'SHOW' toggle). The 'USER TYPE' dropdown menu is highlighted with a red box and currently shows 'System Administrator'. At the bottom right are 'CANCEL' and 'SAVE' buttons.

依次点击“DETAILS”后面的“ORGANIZATIONS”、“TEAMS”、“GRANTED PERMISSIONS”，可以看到该用户拥有对访问所有对象的权限，如下：

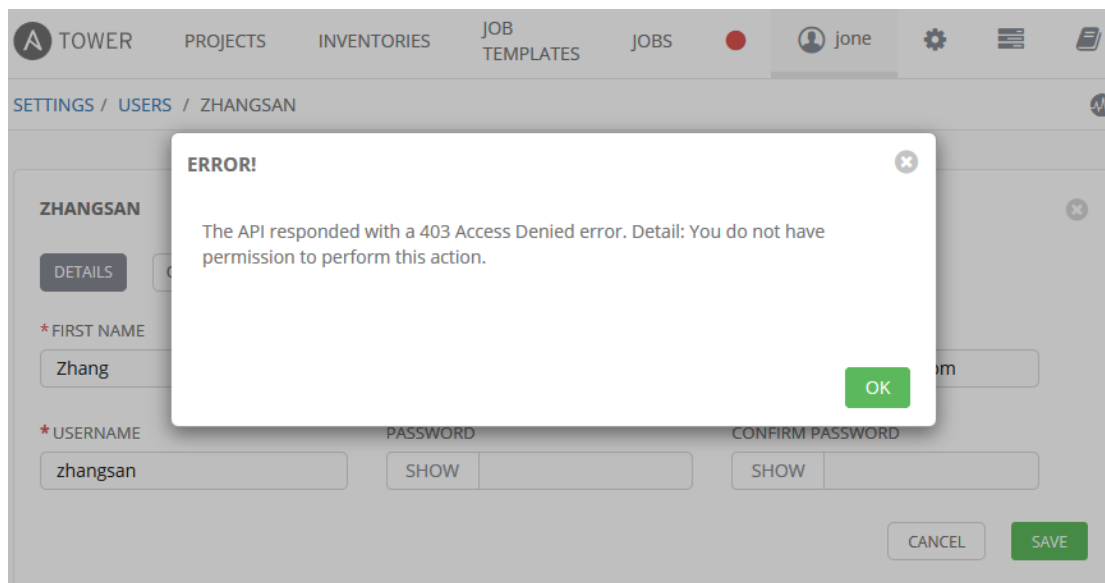
The screenshot shows the 'ORGANIZATIONS' tab for the 'REDHAT ADMIN' user. At the top, there is a search bar with a 'NAME' dropdown and a 'SEARCH' button. Below the search bar, a large gray message box contains the text 'SYSTEM ADMINISTRATORS HAVE ACCESS TO ALL ORGANIZATIONS', which is highlighted with a red box.



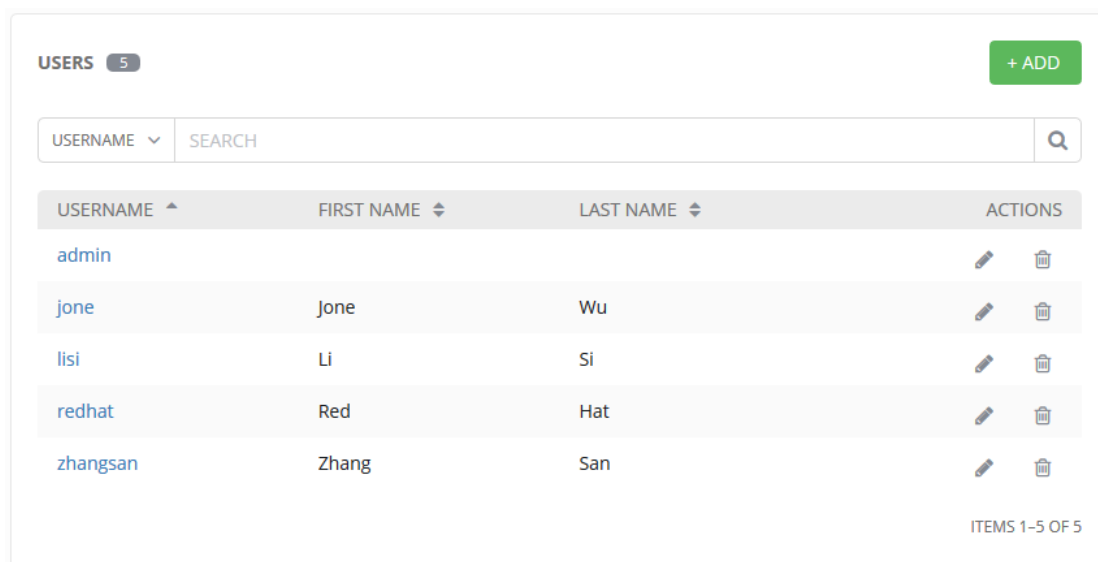
普通用户，系统审计员都没有对用户操作的权限，包括对用户自己的信息进行编辑修改。例如：Tower 中有一个系统用户 jone，用户类型为系统审计员，使用 jone 用户登录 Tower，查看用户信息如下：



使用 jone 用户对其他用户信息进行编辑，提交保存的时候会提示没有权限进行这个操作，如下：



在创建新用户的时候，可以选择用户类型，如需修改已经存在的用户的类型，需使用系统管理员用户登录操作，点击“Settings”→“USERS”，如下：



点击需要编辑的用户后面的第一个编辑按钮，例如：编辑用户 lisi，如下：

LISI

DETAILS ORGANIZATIONS TEAMS GRANTED PERMISSIONS

* FIRST NAME: Li

* LAST NAME: Si

* EMAIL: lisi@example.com

* USERNAME: lisi

PASSWORD: SHOW

CONFIRM PASSWORD: SHOW

USER TYPE: Normal User

CANCEL SAVE

点击用户类型下面的下拉按钮，选择用户类型，如下：

LISI AUDITOR

DETAILS ORGANIZATIONS TEAMS GRANTED PERMISSIONS

* FIRST NAME: Li

* LAST NAME: Si

* EMAIL: lisi@example.com

* USERNAME: lisi

PASSWORD: SHOW

CONFIRM PASSWORD: SHOW

USER TYPE: System Auditor

CANCEL SAVE

点击“SAVE”保存即可。

7.11.3 组织权限

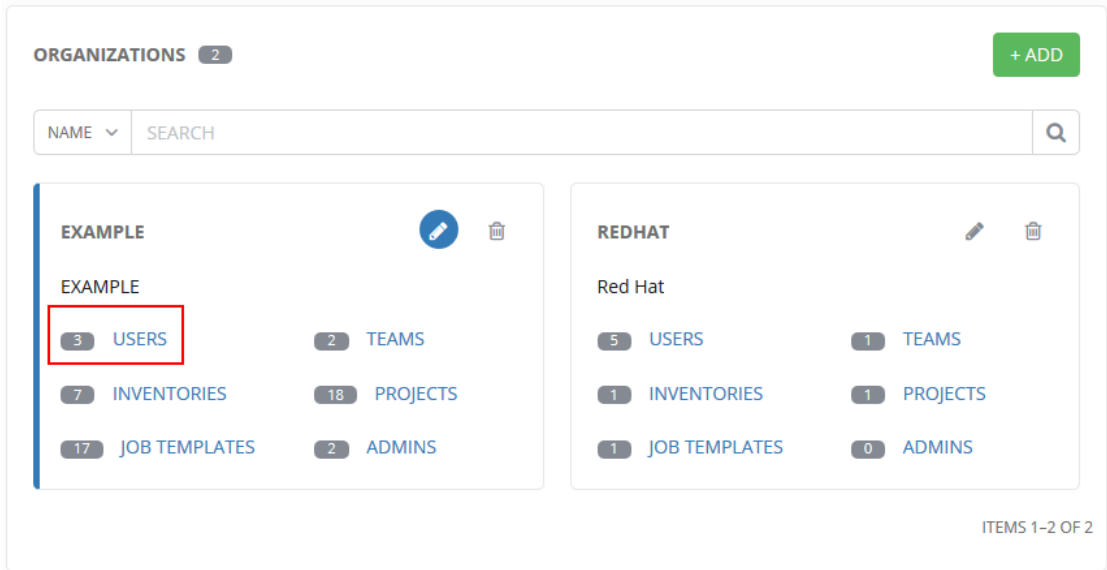
系统管理员和组织管理员可以为组织添加用户、删除用户、分配和回收组织用户角色。

一个普通的系统用户可以被分配组织审计员和组织管理角色，被分配组织审计员角色后，用户可以对指定的组织具有审计的权限，但是对其他组织则没有；被分配组织管理员角色后，用户可以对指定组织具有管理权限，对其他组织则没有。

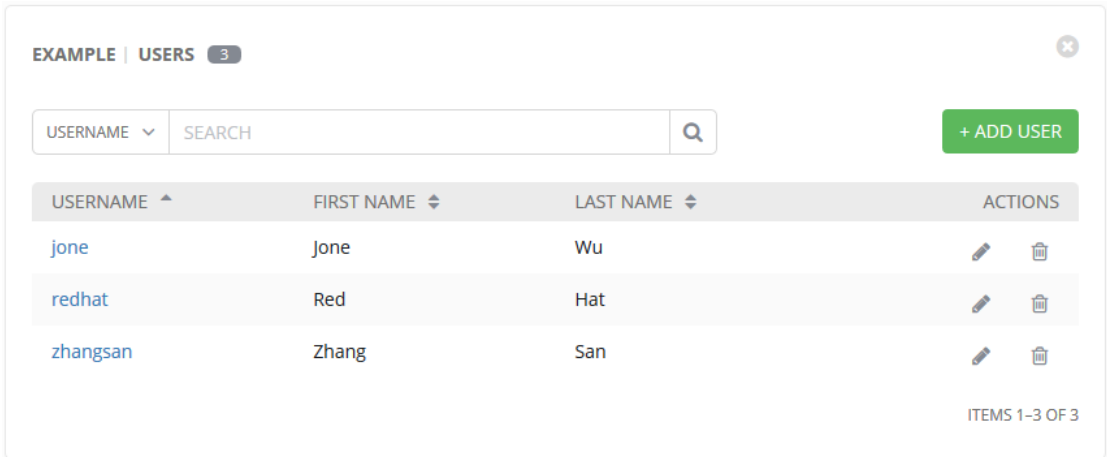
一个用户在创建的时候会被要求加入一个组织，如果用户类型为普通用户，那么这个用户为被分配组织成员的角色，在创建之后，系统管理员可以将这个用户加入到任何组织，组织管理员可以将这个用户加入到自己的组织。

查看组织中的用户

点击“Settings”→“ORGANIZATION”，点击需要查看的组织下的“USERS”，如下：



组织 EXAMPLE 中的用户如下：



点击右侧的“+ADD USER”可以为组织添加用户，添加进来的用户会被分配组织成员角色，如果用户本身已经拥有对组织的其他角色（系统审计员，系统管理员等），那么会以最高权限为准。

查看组织中的管理员

点击“Settings”→“ORGANIZATION”，点击需要查看的组织下的“ADMINS”，如下：

ORGANIZATIONS 2

+ ADD

NAME SEARCH

Q

EXAMPLE

EXAMPLE

4

USERS

2

TEAMS

7

INVENTORIES

18

PROJECTS

17

JOB TEMPLATES

2

ADMINS

REDHAT

Red Hat

5

USERS

1

TEAMS

1

INVENTORIES

1

PROJECTS

1

JOB TEMPLATES

0

ADMINS

ITEMS 1-2 OF 2

组织 EXAMPLE 中的管理员如下：

EXAMPLE | ADMINS 2

×

USERNAME SEARCH

Q

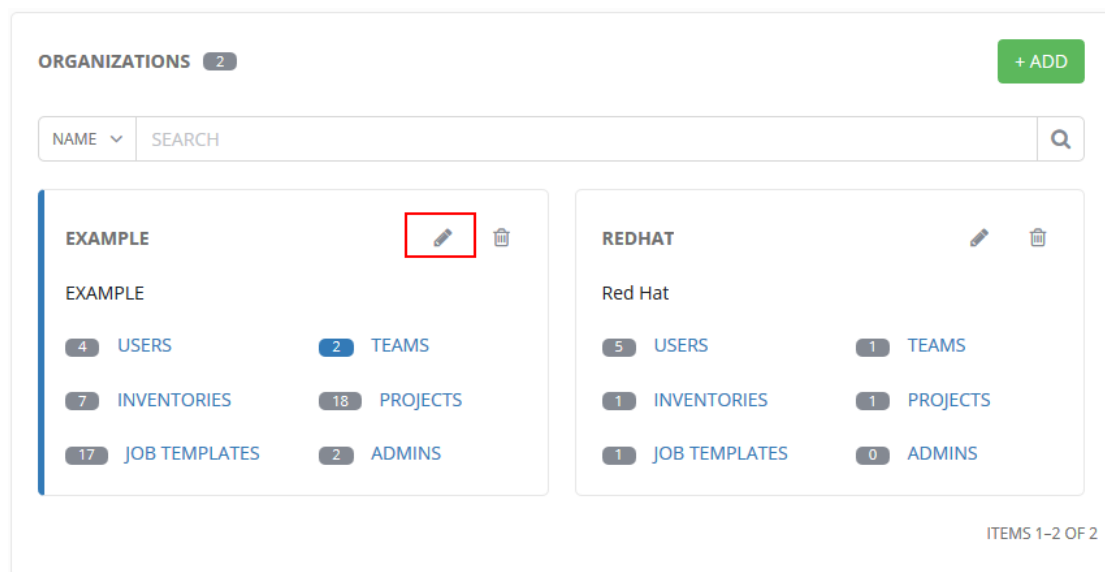
+ ADD ADMINISTRATOR

USERNAME	FIRST NAME	LAST NAME	ACTIONS
lisi	Li	Si	<div><div></div><div></div></div>
zhangsan	Zhang	San	<div><div></div><div></div></div>

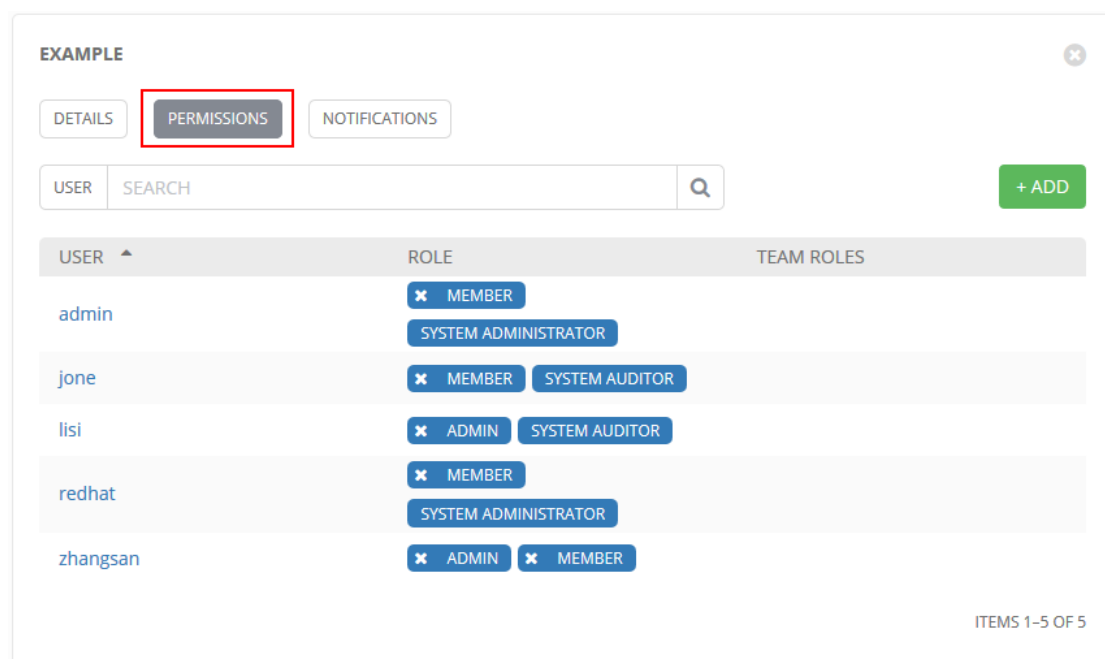
ITEMS 1-2 OF 2

点击右侧的“+ADD ADMINISTRATOR”可以为组织添加管理员，添加进来的用户会被分配组织管理员角色，如果用户本身已经拥有对组织的其他角色（成员，系统审计员等），那么会以最高权限为准。

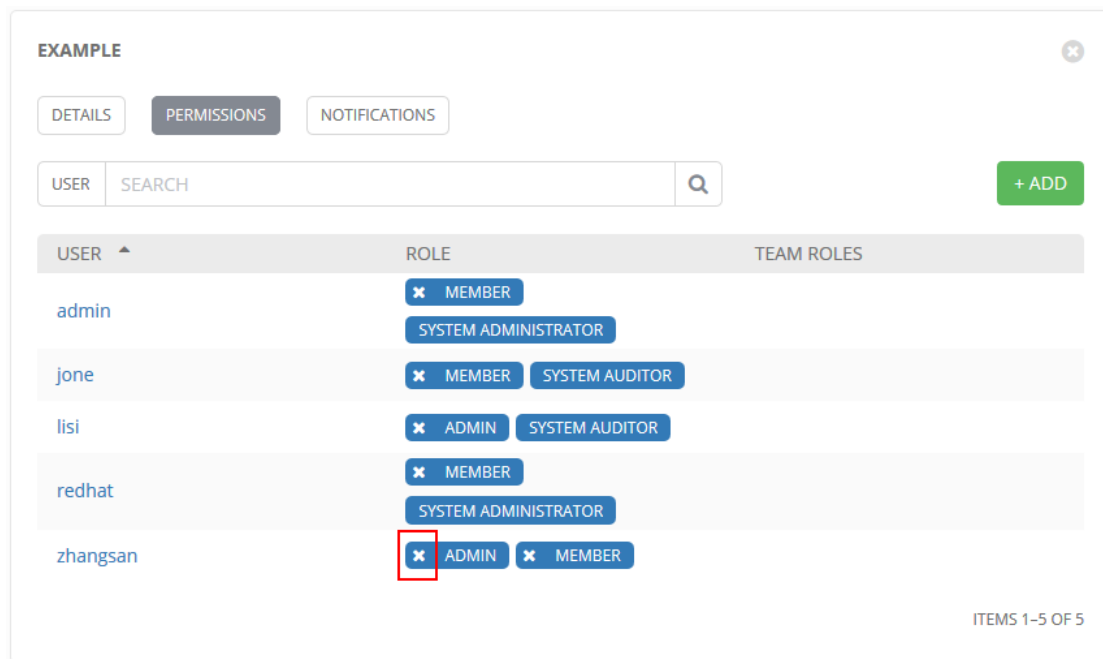
在组织的编辑页面也可以为组织添加、删除用户，以及角色权限的分配管理，点击“Settings” → “ORGANIZATION”，点击需要操作的组织名称后面的编辑按钮，如下：



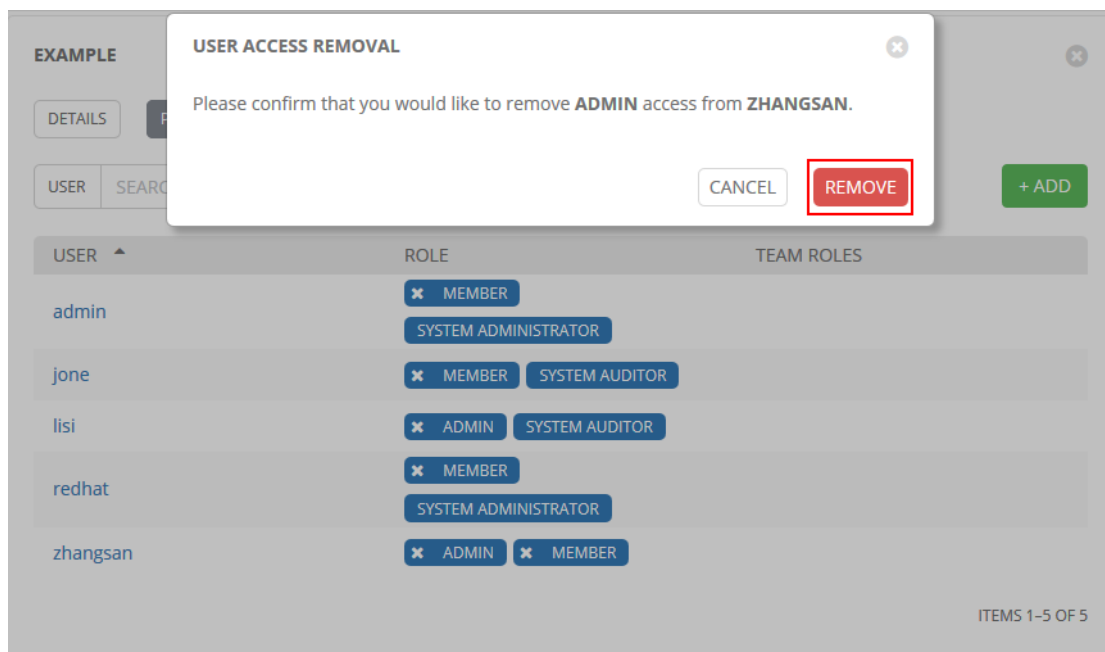
点击“DETAILS”右侧的“PERMISSIONS”，可以看到组织中的用户及每个用户已经分配的角色，如下：



点击用户右侧角色上的“✕”，可以对用户的角色进行回收，例如，回收用户 zhangsan 的管理员角色，点击用户 zhangsan 右侧“ADMIN”角色上的“✕”，如下：



在弹出的对话框上点击“REMOVE”，用户 zhangsan 的管理员权限就被回收，如下：



注：如果用户只有一个角色，若将这个角色回收，那么这个用户就从组织中移除了。

点击“+ADD”可以为组织添加用户和分配权限，例如，添加一个用户，并分配管理员角色，勾选需要添加的用户，在下方选择“Admin”角色，如下：

EXAMPLE | ADD PERMISSIONS



1 Please select Users / Teams from the lists below.

USERS

TEAMS

FIRST NAME ▾

SEARCH

Q

	FIRST NAME ▾	LAST NAME ▾	USERNAME ▲
<input type="checkbox"/>	Jone	Wu	jone
<input type="checkbox"/>	Li	Si	lisi
<input checked="" type="checkbox"/>	Cristiano	Ronaldo	ronaldo
<input type="checkbox"/>	Zhang	San	zhangsan

ITEMS 1-4 OF 4

2 Please assign roles to the selected users/teams

KEY

Cristiano Ronaldo USER

× Admin



CANCEL

SAVE

注：点击“USERS”右侧的“TEAMS”，可以为组织添加团队，对整个团队分配角色，团队中的所有用户都会继承团队的角色。

点击保存，如下：

EXAMPLE

DETAILS

PERMISSIONS

NOTIFICATIONS

USER

SEARCH

Q

+ ADD

USER	ROLE	TEAM ROLES
admin	<div>✕ MEMBER</div> <div>SYSTEM ADMINISTRATOR</div>	
jone	<div>✕ MEMBER</div> <div>SYSTEM AUDITOR</div>	
lisi	<div>✕ ADMIN</div> <div>SYSTEM AUDITOR</div>	
redhat	<div>✕ MEMBER</div> <div>SYSTEM ADMINISTRATOR</div>	
ronaldo	<div>✕ ADMIN</div>	
zhangsan	<div>✕ MEMBER</div>	

ITEMS 1-6 OF 6

7.11.4 项目权限

系统管理员、项目管理员和项目所在组织的组织管理员可以为项目添加用户、删除用户、分配和回收项目用户角色。

项目的角色有三种：

- ADMIN：管理员，对项目的所有对象都有管理权限
- USE：使用，可以在作业模板中使用项目
- UPDATE：更新，可以使用 SCM 更新项目、清单、组

系统管理员对所有项目都有管理权限，组织管理员对组织中的所有项目有管理权限，项目管理员对所在的项目有管理权限。组织中拥有成员角色的用户对组织中的所有项目具有只读权限。

例如，项目 soe 属于组织 EXAMPLE，查看项目 soe 默认分配的权限，点击左上方的“PROJECTS”，在项目列表中查找 soe（如果项目较多可以在搜索框搜索），如下：

TOWER

PROJECTS

INVENTORIES

JOB
TEMPLATES

JOB

admin

PROJECTS

PROJECTS 1

+ ADD

NAME SEARCH

Q

Name : soe

NAME TYPE ACTIONS

soe

Manual

ITEMS 1-1 OF 1

点击项目“soe”，点击“DETAILS”右侧的“PERMISSIONS”，如下：

SOE

DETAILS

PERMISSIONS

NOTIFICATIONS

USER SEARCH

Q

+ ADD

USER ROLE TEAM ROLES

admin

SYSTEM ADMINISTRATOR

jone

SYSTEM AUDITOR

lisi

ADMIN

SYSTEM AUDITOR

redhat

SYSTEM ADMINISTRATOR

ronaldo

ADMIN

ITEMS 1-5 OF 5

项目 soe 已经有 5 个用户分配了角色，其中 admin 和 redhat 是系统管理员，lisi 和 ronaldo 是组织管理员，jone 是系统审计员。

点击右侧的“+ADD”可以为项目添加新的用户，例如，添加用户 zhangsan，并分配 USE 和 UPDATE 角色，点击“+ADD”，如下：

SOE | ADD PERMISSIONS



1 Please select Users / Teams from the lists below.

USERS

TEAMS

FIRST NAME ▾

SEARCH



	FIRST NAME ▴ ▾	LAST NAME ▴ ▾	USERNAME ▲
<input type="checkbox"/>	Jone	Wu	jone
<input type="checkbox"/>	Li	Si	lisi
<input type="checkbox"/>	Cristiano	Ronaldo	ronaldo
<input type="checkbox"/>	Zhang	San	zhangsan

ITEMS 1-4 OF 4

CANCEL

SAVE

勾选用户“zhangsan”，点击下方下拉框选择需要分配的角色，如下：

SOE | ADD PERMISSIONS



1 Please select Users / Teams from the lists below.

USERS

TEAMS

FIRST NAME ▾

SEARCH



	FIRST NAME ▴ ▾	LAST NAME ▴ ▾	USERNAME ▲
<input type="checkbox"/>	Jone	Wu	jone
<input type="checkbox"/>	Li	Si	lisi
<input type="checkbox"/>	Cristiano	Ronaldo	ronaldo
<input checked="" type="checkbox"/>	Zhang	San	zhangsan

ITEMS 1-4 OF 4

2 Please assign roles to the selected users/teams

KEY

Zhang San USER

✕ Use

✕ Update

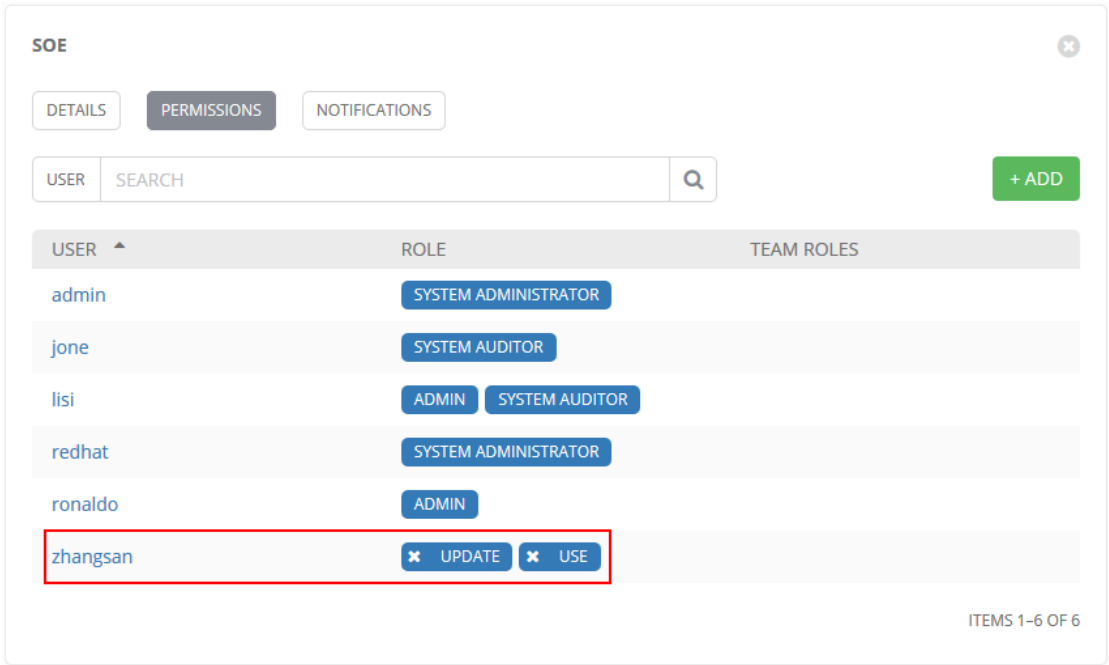


CANCEL

SAVE

注：点击“USERS”右侧的“TEAMS”，可以为项目添加团队，对整个团队分配角色，团队中的所有用户都会继承团队的角色。

点击“SAVE”保存，如下：



用户 zhangsan 被分配了 USE 和 UPDATE 角色，同时点击用户角色前面的“×”，可以对角色进行回收。

7.11.5 清单权限

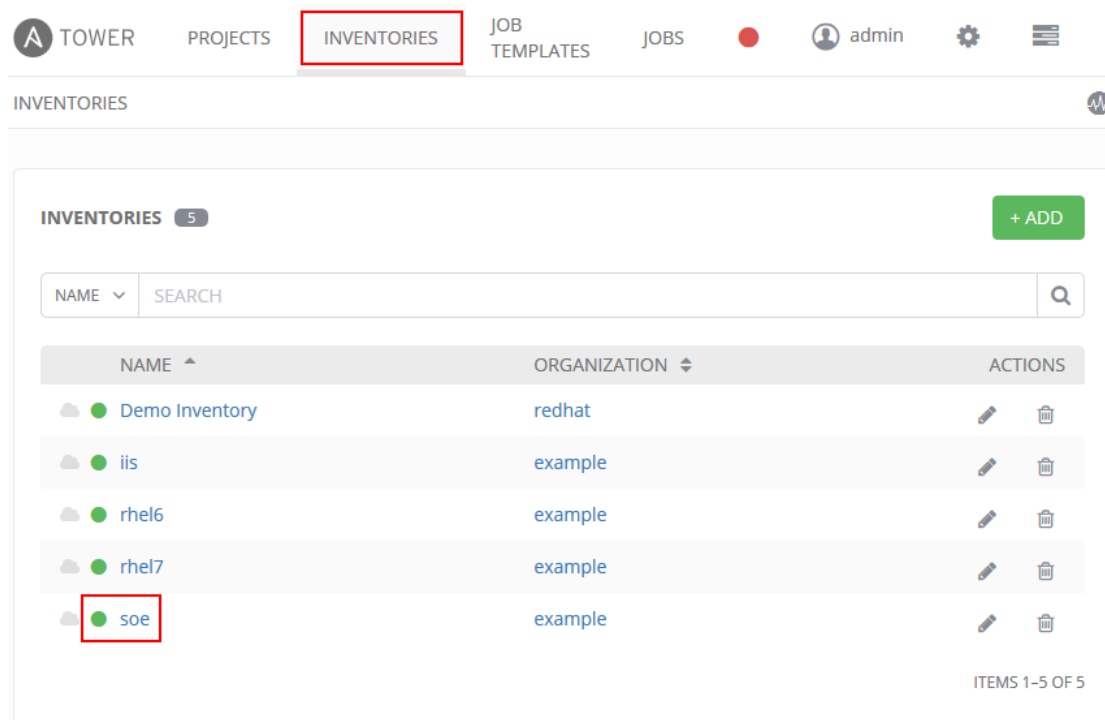
系统管理员可以为任何组织创建清单，组织管理员只能为所在的组织创建清单。系统管理员可以为任何组织中的清单添加、删除用户，分配角色，组织管理员只能为所在的组织中的清单添加、删除用户、分配角色。

清单的角色有四种：

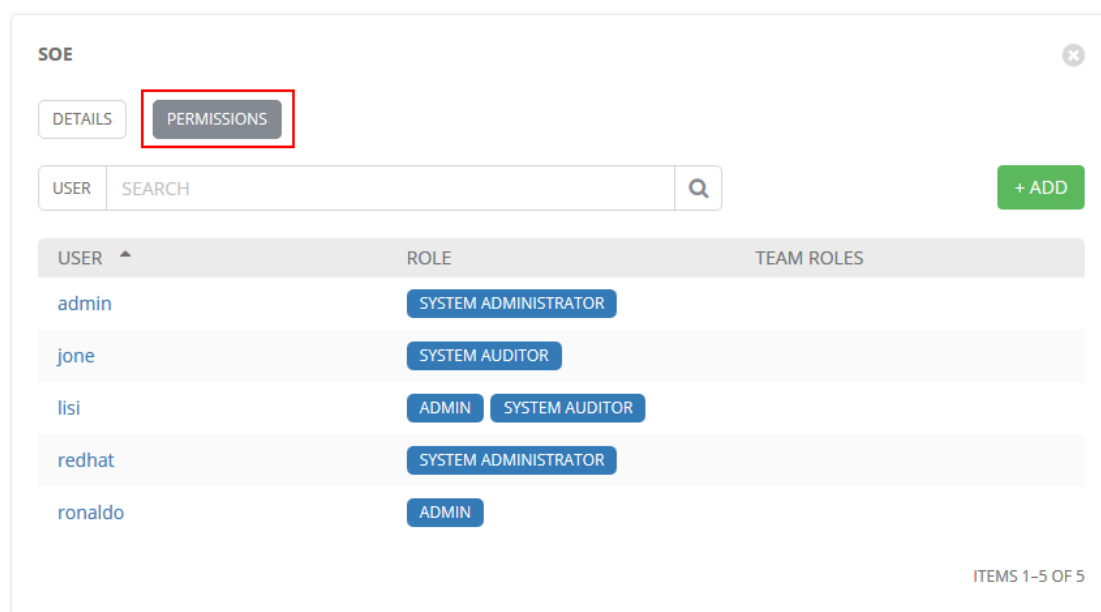
- ADMIN：管理员，对清单的所有对象都有管理权限
- USE：使用，可以在作业模板中使用清单
- UPDATE：更新，可以使用 SCM 更新项目、清单、组
- AD HOC：临时，可以在清单上运行临时命令（有限数量的命令）

系统管理员对所有清单都有管理权限，组织管理员对组织中的所有清单有管理权限，清单管理员对所在的清单有管理权限。组织中拥有成员角色的用户对组织中的所有清单具有只读权限。

例如，清单 soe 属于组织 EXAMPLE，查看清单 soe 默认分配的权限，点击左上方的“INVENTORIES”，在清单列表中查找 soe（如果清单较多可以在搜索框搜索），如下：



点击清单“soe”右侧的编辑按钮，再点击“DETAILS”右侧的“PERMISSIONS”，如下：



清单 soe 已经有 5 个用户分配了角色，其中 admin 和 redhat 是系统管理员，lisi 和 ronaldo 是组织管理员，jone 是系统审计员。

点击右侧的“+ADD”可以为清单添加新的用户，例如，添加用户 zhangsan，并分配 USE、UPDATE 和 AD HOC 角色，点击“+ADD”，如下：

SOE | ADD PERMISSIONS



1 Please select Users / Teams from the lists below.

USERS

TEAMS

FIRST NAME ▾

SEARCH



	FIRST NAME ▴ ▾	LAST NAME ▴ ▾	USERNAME ▲
<input type="checkbox"/>	Jone	Wu	jone
<input type="checkbox"/>	Li	Si	lisi
<input type="checkbox"/>	Cristiano	Ronaldo	ronaldo
<input type="checkbox"/>	Zhang	San	zhangsan

ITEMS 1-4 OF 4

CANCEL

SAVE

勾选用户“zhangsan”，点击下方下拉框选择需要分配的角色，如下：

SOE | ADD PERMISSIONS



1 Please select Users / Teams from the lists below.

USERS

TEAMS

FIRST NAME ▾

SEARCH



	FIRST NAME ▴ ▾	LAST NAME ▴ ▾	USERNAME ▲
<input type="checkbox"/>	Jone	Wu	jone
<input type="checkbox"/>	Li	Si	lisi
<input type="checkbox"/>	Cristiano	Ronaldo	ronaldo
<input checked="" type="checkbox"/>	Zhang	San	zhangsan

ITEMS 1-4 OF 4

2 Please assign roles to the selected users/teams

KEY

Zhang San USER

✕ Use

✕ Ad Hoc

✕ Update

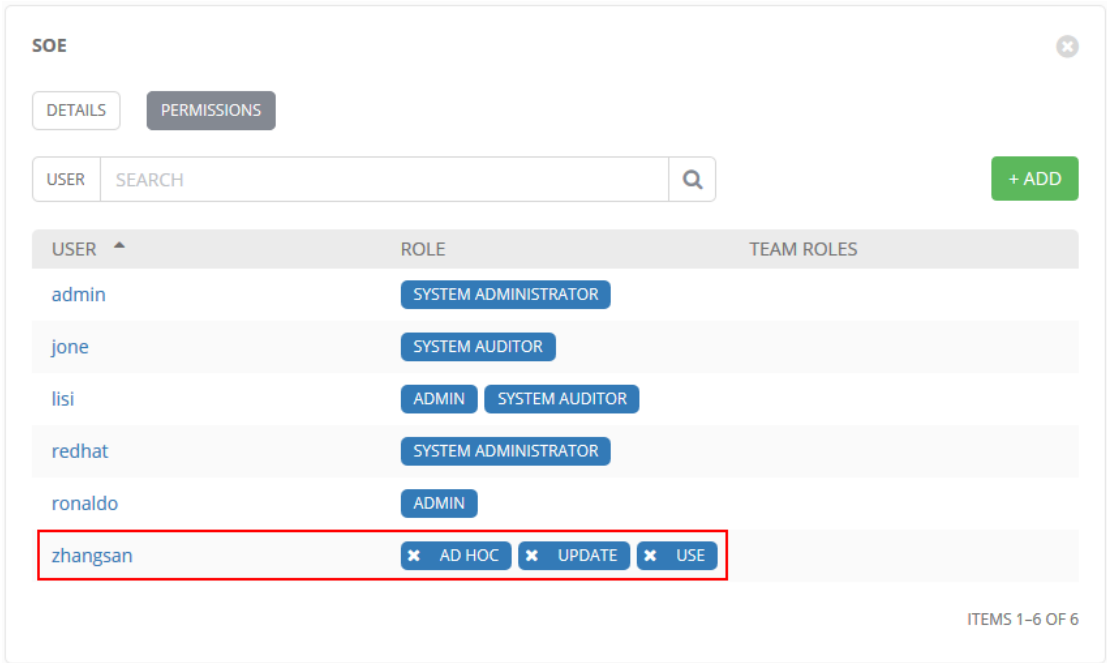


CANCEL

SAVE

注：点击“USERS”右侧的“TEAMS”，可以为清单添加团队，对整个团队分配角色，团队中的所有用户都会继承团队的角色。

点击“SAVE”保存，如下：



用户 zhangsan 被分配了 USE、UPDATE 和 AD HOC 角色，同时点击用户角色前面的“×”，可以对角色进行回收。

7.11.6 凭证权限

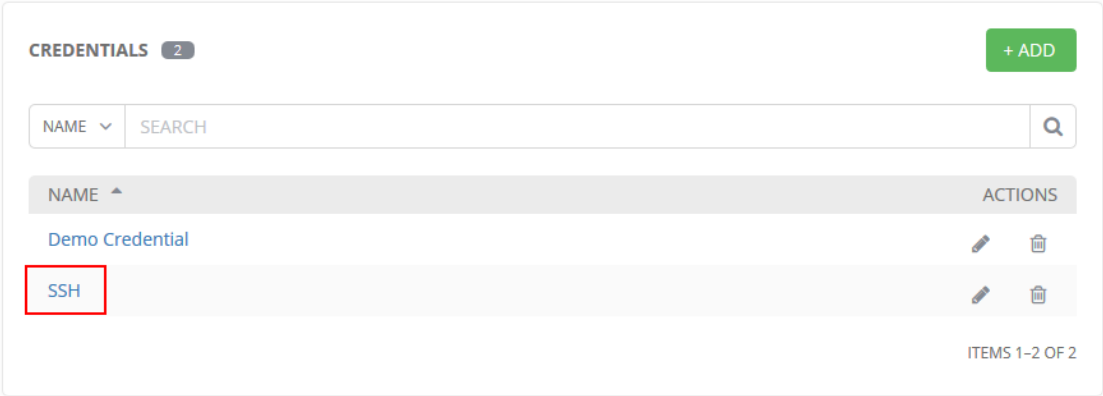
系统管理员可以为任何组织创建凭证，组织管理员只能为所在的组织创建凭证。系统管理员可以为任何组织中的凭证添加、删除用户，分配角色，组织管理员只能为所在的组织中的凭证添加、删除用户、分配角色。

凭证的角色有两种：

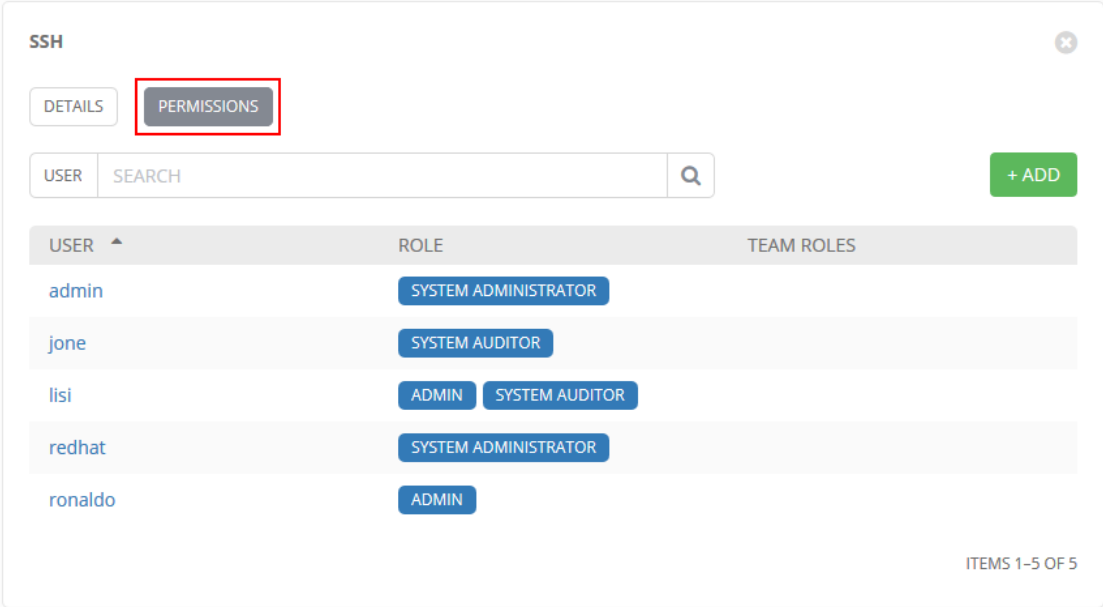
- ADMIN：管理员，对凭证的所有对象都有管理权限
- USE：使用，可以在作业模板中使用凭证

系统管理员对所有凭证都有管理权限，组织管理员对组织中的所有凭证有管理权限，凭证管理员对所在的凭证有管理权限。组织中拥有成员角色的用户对组织中的所有凭证具有只读权限。

例如，凭证 SSH 属于组织 EXAMPLE，查看凭证 SSH 默认分配的权限，点击“Settings”→“CREDENTIALS”，在凭证列表中查找 SSH（如果清单较多可以在搜索框搜索），如下：



点击凭证“SSH”，再点击“DETAILS”右侧的“PERMISSIONS”，如下：



凭证 SSH 已经有 5 个用户分配了角色，其中 admin 和 redhat 是系统管理员，lisi 和 ronaldo 是组织管理员，jone 是系统审计员。

点击右侧的“+ADD”可以为凭证添加新的用户，例如，添加用户 zhangsan，并分配 USE 角色，点击“+ADD”，如下：

SSH | ADD PERMISSIONS



1 Please select Users / Teams from the lists below.

USERS

TEAMS

FIRST NAME ▾

SEARCH



	FIRST NAME ▴ ▾	LAST NAME ▴ ▾	USERNAME ▲
<input type="checkbox"/>	Jone	Wu	jone
<input type="checkbox"/>	Li	Si	lisi
<input type="checkbox"/>	Cristiano	Ronaldo	ronaldo
<input type="checkbox"/>	Zhang	San	zhangsan

ITEMS 1-4 OF 4

CANCEL

SAVE

勾选用户“zhangsan”，点击下方下拉框选择需要分配的角色，如下：

SSH | ADD PERMISSIONS



1 Please select Users / Teams from the lists below.

USERS

TEAMS

FIRST NAME ▾

SEARCH



	FIRST NAME ▴ ▾	LAST NAME ▴ ▾	USERNAME ▲
<input type="checkbox"/>	Jone	Wu	jone
<input type="checkbox"/>	Li	Si	lisi
<input type="checkbox"/>	Cristiano	Ronaldo	ronaldo
<input checked="" type="checkbox"/>	Zhang	San	zhangsan

ITEMS 1-4 OF 4

2 Please assign roles to the selected users/teams

KEY

Zhang San USER

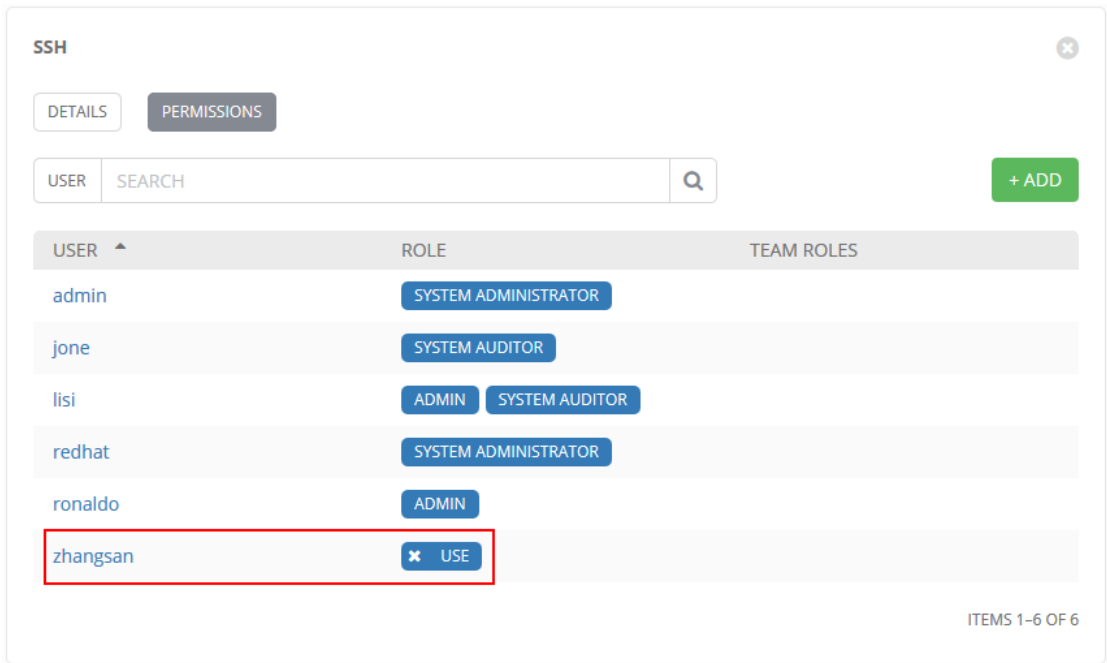


CANCEL

SAVE

注：点击“USERS”右侧的“TEAMS”，可以为清单添加团队，对整个团队分配角色，团队中的所有用户都会继承团队的角色。

点击“SAVE”保存，如下：



用户 zhangsan 被分配了 USE 角色，同时点击用户角色前面的“×”，可以对角色进行回收。

7.11.7 作业模板权限

系统管理员、模板所在组织的组织管理员、模板所在项目的项目管理员和模板管理员可以为模板添加用户、删除用户、分配和回收模板用户角色。

模板的角色有两种：

- ADMIN： 管理员，对模板的所有对象都有管理权限
- EXECUTE： 执行，可以执行作业模板

系统管理员对所有模板都有管理权限，组织管理员对组织中所有项目中的模板有管理权限，项目管理员对项目中的所有模板有管理权限，模板管理员对所在的模板有管理权限。组织中拥有成员角色的用户对组织中所有项目中的模板具有只读权限。

例如，模板 system-config-soe 属于项目 soe，项目 soe 属于组织 EXAMPLE，查看模板 system-config-soe 默认分配的权限，点击左上方的“JOB TEMPLATES”，在模板列表中查找 system-config-soe（如果模板较多可以在搜索框搜索），如下：






TOWER PROJECTS INVENTORIES **JOB TEMPLATES** JOBS admin

JOB TEMPLATES

JOB TEMPLATES 1 + ADD

NAME SEARCH

Name : system-config-soe

NAME	ACTIONS
system-config-soe	    


ITEMS 1-1 OF 1

点击模板“system-config-soe”，点击“DETAILS”右侧的“PERMISSIONS”，如下：

SYSTEM-CONFIG-SOE

DETAILS COMPLETED JOBS **PERMISSIONS** NOTIFICATIONS

USER SEARCH + ADD

USER	ROLE	TEAM ROLES
admin	 ADMIN SYSTEM ADMINISTRATOR	
jone	SYSTEM AUDITOR	
lisi	ADMIN SYSTEM AUDITOR	
redhat	SYSTEM ADMINISTRATOR	
ronaldo	ADMIN	

ITEMS 1-5 OF 5

模板 system-config-soe 已经有 5 个用户分配了角色，其中 admin 和 redhat 是系统管理员，lisi 和 ronaldo 是组织管理员，jone 是系统审计员。

点击右侧的“+ADD”可以为模板添加新的用户，例如，添加用户 zhangsan，并分配 EXECUTE 角色，点击“+ADD”，如下：



1 Please select Users / Teams from the lists below.

USERS

TEAMS

FIRST NAME

SEARCH

	FIRST NAME	LAST NAME	USERNAME
<input type="checkbox"/>	Jone	Wu	jone
<input type="checkbox"/>	Li	Si	lisi
<input type="checkbox"/>	Cristiano	Ronaldo	ronaldo
<input type="checkbox"/>	Zhang	San	zhangsan

ITEMS 1-4 OF 4

CANCEL

SAVE

勾选用户“zhangsan”，点击下方下拉框选择需要分配的角色，如下：



1 Please select Users / Teams from the lists below.

USERS

TEAMS

FIRST NAME

SEARCH

	FIRST NAME	LAST NAME	USERNAME
<input type="checkbox"/>	Jone	Wu	jone
<input type="checkbox"/>	Li	Si	lisi
<input type="checkbox"/>	Cristiano	Ronaldo	ronaldo
<input checked="" type="checkbox"/>	Zhang	San	zhangsan

ITEMS 1-4 OF 4

2 Please assign roles to the selected users/teams

KEY

Zhang San USER

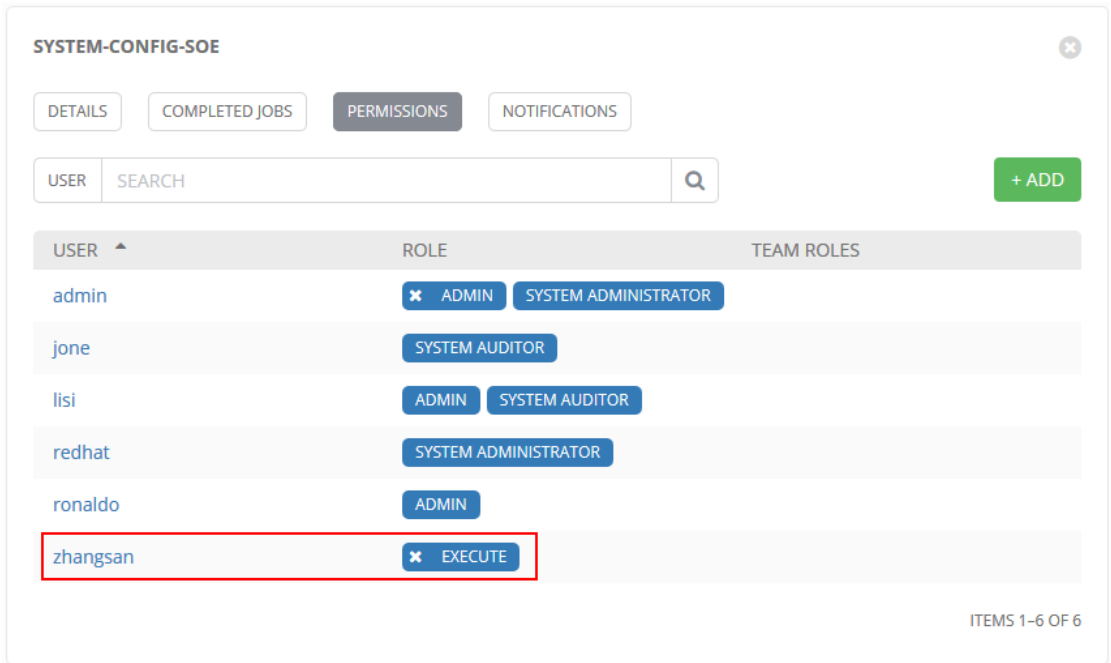
Execute

CANCEL

SAVE

注：点击“USERS”右侧的“TEAMS”，可以为模板添加团队，对整个团队分配角色，团队中的所有用户都会继承团队的角色。

点击“SAVE”保存，如下：



用户 zhangsan 被分配了 EXECUTE 角色，同时点击用户角色前面的“×”，可以对角色进行回收。

7.11.8 只读权限

一个用户若没有分配任何角色，这个用户只能查看所在组织中的对象，对其他组织没有访问权限，若这个用户拥有一个组织中的某一个项目的成员角色，那么这个用户对这个项目有访问权限，对这个项目的组织和这个组织中的其他项目没有访问权限，这个用户可以创建自己的凭证。

7.11.9 内置角色列表

下表列出了基于角色访问控制的系统角色以及在 Tower 中的定义的简短描述。

序号	角色	范围	权限
1	System Administrator	系统范围	管理所有对象
2	System Auditor	系统范围	浏览所有对象内容
3	Ad Hoc Role	清单	在清单上运行临时命令

4	Admin Role	组织、项目、清单、作业模板	管理指定的组织、项目、清单、作业模板
5	Auditor Role	组织、项目、清单、作业模板	浏览指定的组织、项目、清单、作业模板内容
6	Execute Role	作业模板	运行作业模板
7	Member Role	组织、团队	属于指定组织、团队的成员
8	Read Role	所有	浏览指定的组织、项目、清单、作业模板的设置
9	SCM Update Role	项目	使用 SCM 更新项目
10	Update Role	清单	更新云清单
11	Owner Role	凭证	拥有和管理凭证
12	Use Role	凭证、清单、项目	在作业模板中使用凭证、清单、项目

8. 配置 WINDOWS 被 ANSIBLE 管理

从 1.7 版本开始, Ansible 也开始支持 Windows 机器的管理。不过是通过本机的 PowerShell 来实现远程管理, 而不是 SSH。使用 Python 的 “winrm” 模块来和远程 Windows 主机交互。

在管理的过程中, Ansible 无需在远程 Win 主机上安装任何额外的软件, 仍然使用 agentless(非 c/s 架构)。

8.1 Ansible 安装 winrm 模块

Ansible 需要管理 Windows, 需要在 Ansible 上安装 winrm 模块。

模块下载地址: <http://github.com/diyan/pywinrm/archive/master.zip>

该模块在安装的过程中可能还依赖一些其他的模块, 如果在环境允许的情况下可以让 Ansible 管理端连接互联网自动下载依赖, 否则需要手动下载。

安装方法如下:

```
下载 winrm 模块
# wget http://github.com/diyan/pywinrm/archive/master.zip
解压
# unzip pywinrm-master
进入模块目录
# cd pywinrm-master
安装
# python setup.py install
查询是否安装成功
# python -c "import winrm"
没有输出, 表示成功
```

8.2 配置 Windows PowerShell

为了 Ansible 能管理 Windows 机器, 须开启并配置远程 Windows 机器上 PowerShell。

推荐使用脚本自动化配置, 脚本下载地址:

<https://github.com/ansible/ansible/blob/devel/examples/scripts/ConfigureRemoteNodes.ps1>

注: Windows 7 和 Server 2008 R2 系统因为 Windows Management Framework 3.0 的 BUG, 必须安装 hotfix <http://support.microsoft.com/kb/2842230> 来避免内存溢出 (OOM) 和堆

栈异常。新安装的 Server 2008 R2 系统没有升级到最新版本的均存在这个问题。Windows 8.1 and Server 2012 R2 不受影响是因为他们自身默认使用的是 Windows Management Framework 4.0。

以下所有操作都必须使用 administrator 账号执行。

下载脚本后，将脚本拷贝至远程 Windows 机器上，运行方法有两种：

方法一：直接在脚本上点击右键，选择“使用 PowerShell 运行”



方法二：在 PowerShell 命令行里运行

```
PS C:\Users\Administrator> cd d:\
PS D:\> ls

目录: D:\

Mode                LastWriteTime         Length Name
----                -
-a---             2017/1/17  9:51         10167 ConfigureRemotingForAnsible.ps1

PS D:\> .\ConfigureRemotingForAnsible.ps1 -Verbose
详细信息: Verifying WinRM service.
详细信息: PS Remoting is already enabled.
详细信息: SSL listener is already active.
详细信息: Basic auth is already enabled.
详细信息: Firewall rule already exists to allow WinRM HTTPS.
详细信息: HTTP: Enabled | HTTPS: Enabled
详细信息: PS Remoting has been successfully configured for Ansible.
```

注：可能有些机器的 PowerShell 在运行脚本的时候会出现以下错误：

```
PS D:\> .\ConfigureRemotingForAnsible.ps1
.\ConfigureRemotingForAnsible.ps1 : 无法加载文件 D:\ConfigureRemotingForAnsible.ps1，因为在此系统上禁止运行脚本。有关详
细信息，请参阅 http://go.microsoft.com/fwlink/?LinkID=135170 中的 about_Execution_Policies。
所在位置 行:1 字符: 1
+ ~~~~~
+ .\ConfigureRemotingForAnsible.ps1
+ ~~~~~
+ CategoryInfo          : SecurityError: (:) [], PSSecurityException
+ FullyQualifiedErrorId : UnauthorizedAccess
```

这种情况是因为 PowerShell 的执行策略被设置为 Restricted，该策略禁止运行任何脚本，将执行策略修改为 RemoteSigned 即可，检查及修改方法如下：

```
PS D:\> get-executionpolicy
Restricted
PS D:\> set-executionpolicy remotesigned

执行策略更改
执行策略可以防止您执行不信任的脚本。更改执行策略可能会使您面临安全风险。
帮助主题中所述的安全风险。是否要更改执行策略?
[Y] 是(Y) [N] 否(N) [S] 挂起(S) [?] 帮助 (默认值为“Y”): y
PS D:\> get-executionpolicy
RemoteSigned
```

多数 Ansible Windows 模块需要 PowerShell 3.0 或更高版本，同时也需要在其基础上运行安装脚本。需要注意的是 PowerShell 3.0 只在 Windows 7 SP1, Windows Server 2008 SP1, 和更新的 windows 发布版才被支持。

如需更新到 PowerShell 3.0，可以使用以下的连接来下载脚本进行更新：

https://github.com/cchurch/ansible/blob/devel/examples/scripts/upgrade_to_powershell3.ps1

8.3 连接测试

创建一个 Inventory 文件，内容如下：

```
# vim win
[test]
172.168.0.35 ansible_ssh_user=administrator ansible_ssh_pass=xxxxxx
ansible_ssh_port=5986 ansible_connection=winrm
```

参数说明：

ansible_ssh_user 表示远程 windows 机器上的用户

ansible_ssh_pass 表示 ansible_ssh_user 对应用户的密码

ansible_ssh_port 表示连接 PowerShell 使用的端口，这里是 5986

ansible_connection 表示连接 windows 的类型，这里不能使用 ssh

测试，使用 ping 模块测试连通性，windows 使用的是 win_ping

```
# ansible -i win all -m win_ping
```

```
172.168.0.35 | UNREACHABLE! => {
  "changed": false,
  "msg": "ssl: [SSL: CERTIFICATE_VERIFY_FAILED] certificate verify failed
(_ssl.c:579)",
  "unreachable": true
}
```

失败，因为使用的是自签名证书，所以需要添加以下参数来忽略错误：

```
# vim win

[test]

172.168.0.35 ansible_ssh_user=administrator ansible_ssh_pass=xxxxxx
ansible_ssh_port=5986 ansible_connection=winrm
ansible_winrm_server_cert_validation=ignore
```

再次测试，成功：

```
# ansible -i win all -m win_ping
172.168.0.35 | SUCCESS => {
  "changed": false,
  "ping": "pong"
}
```