|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | | |  |
| **Huawei Ansible Plug-in**  **V2.0.4** | | |  |
| **用户指南** | | |
| **文档版本** | **01** | |
| **发布日期** | **2020-05-06** | |
|  | | | | |
|  | 华为技术有限公司 | |  |  |

|  |
| --- |
| 版权所有 © 华为技术有限公司2020。 保留一切权利。  非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。  商标声明  和其他华为商标均为华为技术有限公司的商标。  本文档提及的其他所有商标或注册商标，由各自的所有人拥有。  注意  您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或默示的声明或保证。  由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。 |

|  |  |
| --- | --- |
| 华为技术有限公司 | |
| 地址： | 深圳市龙岗区坂田华为总部办公楼 邮编：518129 |
| 网址： | <https://e.huawei.com> |

前言

概述

本文档详细的描述了如何安装和卸载华为Ansible插件、以及如何使用插件实现查看华为服务器的信息和健康状态查询、配置、部署、固件升级等功能。

读者对象

本指南主要适用于以下工程师：

* 技术支持工程师
* 系统维护工程师

符号约定

在本文中可能出现下列标志，它们所代表的含义如下。

| **符号** | **说明** |
| --- | --- |
|  | 用于警示紧急的危险情形，若不避免，将会导致人员死亡或严重的人身伤害。 |
|  | 用于警示潜在的危险情形，若不避免，可能会导致人员死亡或严重的人身伤害。 |
|  | 用于警示潜在的危险情形，若不避免，可能会导致中度或轻微的人身伤害。 |
|  | 用于传递设备或环境安全警示信息，若不避免，可能会导致设备损坏、数据丢失、设备性能降低或其它不可预知的结果。  “注意”不涉及人身伤害。 |
|  | 用于突出重要/关键信息、最佳实践和小窍门等。  “说明”不是安全警示信息，不涉及人身、设备及环境伤害。 |

修改记录

| **文档版本** | **发布日期** | **修改说明** |
| --- | --- | --- |
| 01 | 2020-05-06 | 第一次正式发布。 |

目 录

[前言 ii](#_Toc39675598)

[1 简介 1](#_Toc39675599)

[2 安装和卸载华为Ansible插件 3](#_Toc39675600)

[2.1 环境依赖 3](#_Toc39675601)

[2.1.1 软件依赖 3](#_Toc39675602)

[2.1.2 版本配套关系 3](#_Toc39675603)

[2.2 安装华为Ansible插件 4](#_Toc39675604)

[2.3 卸载华为Ansible插件 4](#_Toc39675605)

[3 配置华为Ansible插件 5](#_Toc39675606)

[3.1 配置/etc/ansible/hosts文件 5](#_Toc39675607)

[3.2 配置/group\_vars/myhosts文件 5](#_Toc39675608)

[3.3 配置SSL证书认证和TLS 1.2通讯方式 6](#_Toc39675609)

[4 使用华为Ansible插件 8](#_Toc39675610)

[4.1 查询服务器基本信息（生成json文件） 10](#_Toc39675611)

[4.2 管理设备启动项 11](#_Toc39675612)

[4.2.1 查询启动项配置信息 11](#_Toc39675613)

[4.2.2 配置启动设备 12](#_Toc39675614)

[4.3 管理电源 13](#_Toc39675615)

[4.3.1 查询电源状态 13](#_Toc39675616)

[4.3.2 配置电源状态 13](#_Toc39675617)

[4.4 管理iBMC用户 14](#_Toc39675618)

[4.4.1 查询iBMC用户（生成json文件） 15](#_Toc39675619)

[4.4.2 创建iBMC用户 15](#_Toc39675620)

[4.4.3 修改iBMC用户 17](#_Toc39675621)

[4.4.4 删除iBMC用户 19](#_Toc39675622)

[4.5 iBMC网络配置 20](#_Toc39675623)

[4.5.1 查询iBMC网络信息（生成json文件） 20](#_Toc39675624)

[4.5.2 配置iBMC网络信息 20](#_Toc39675625)

[4.6 管理NTP服务 22](#_Toc39675626)

[4.6.1 查询NTP服务信息 22](#_Toc39675627)

[4.6.2 配置NTP 23](#_Toc39675628)

[4.7 管理SNMP Trap服务 24](#_Toc39675629)

[4.7.1 查询SNMP服务信息（生成json文件） 24](#_Toc39675630)

[4.7.2 配置SNMP Trap 25](#_Toc39675631)

[4.8 导入导出Profile文件 27](#_Toc39675632)

[4.9 固件升级 28](#_Toc39675633)

[4.9.1 查询固件版本信息（生成json文件） 28](#_Toc39675634)

[4.9.2 升级固件 29](#_Toc39675635)

[4.9.2.1 带外固件升级 29](#_Toc39675636)

[4.9.2.2 带内固件升级 30](#_Toc39675637)

[4.10 RAID配置 32](#_Toc39675638)

[4.10.1 查询RAID配置（生成json文件） 32](#_Toc39675639)

[4.10.2 删除RAID组 33](#_Toc39675640)

[4.10.3 创建RAID组 34](#_Toc39675641)

[4.10.4 修改RAID配置 36](#_Toc39675642)

[4.11 OS部署 38](#_Toc39675643)

[4.11.1 ServiceCD2.0方式 38](#_Toc39675644)

[4.11.2 Smart Provisioning方式 42](#_Toc39675645)

[A 如何获取帮助 49](#_Toc39675646)

[A.1 联系华为前准备 49](#_Toc39675647)

[A.2 如何从网站获取帮助 49](#_Toc39675648)

# 简介

华为Ansible插件是一个集成在Ansible软件中的管理华为服务器的插件，通过Redfish接口对接iBMC，该插件可以实现对华为服务器查询、配置、部署、升级等功能。

具体支持以下功能：

* 查询服务器基本信息及健康状态
* 配置服务器的启动设备
* 管理服务器的电源
* 管理iBMC用户
* 查询和配置iBMC网络信息
* 查询和配置NTP服务
* 查询和配置SNMP服务
* 导入或导出服务器的Profile文件
* 升级服务器带外和带内固件
* 查询和配置RAID
* 部署服务器的OS（支持ServiceCD2.0和Smart Provisioning方式）

华为Ansible插件支持的服务器如表1-1所示。

支持的服务器

| 架构 | 类型 | 服务器型号 |
| --- | --- | --- |
| x86 | 机架服务器 | RH2288H V3 |
| 2488 V5 |
| 2288H V5 |
| 刀片服务器 | CH121 V3 |
| CH242 V3 DDR4 |
| CH121 V5 |
| CH242 V5 |
| 高密服务器 | XH622 V3 |
| XH321 V5 |
| Arm | 机架服务器 | TaiShan 200服务器（型号2280） |
| TaiShan 200服务器（型号5280） |
| TaiShan 200服务器（型号5290） |
| TaiShan 200服务器（型号2480） |

# 安装和卸载华为Ansible插件

[2.1 环境依赖](#_ZH-CN_TOPIC_0102786727)

[2.2 安装华为Ansible插件](#_ZH-CN_TOPIC_0102786743)

[2.3 卸载华为Ansible插件](#_ZH-CN_TOPIC_0102786567)

## 环境依赖

### 软件依赖

* sshpass 1.06及以上版本
* Python 2.7.9和python-requests 2.6.0及以上版本
* jq 1.5及以上版本

### 版本配套关系

| 版本名称 | 配套版本 |
| --- | --- |
| Ansible | V2.5.0及以上版本 |
| iBMC | * x86 V5服务器：V325及以上版本 * x86 V3服务器：V323及以上版本 * Arm服务器：   + - 对于版本号为Vxxx格式的iBMC，仅支持V556及以上版本     - 对于版本号为x.xx.xx.xx格式的iBMC，仅支持3.01.01.00及以上版本 |
| BIOS | * x86 V5服务器：V119及以上版本 * x86 V3服务器：V513及以上版本 * Arm服务器：V112及以上版本 |
| Smart Provisioning | * x86服务器：V118及以上版本，可访问[Smart Provisioning](https://support.huawei.com/enterprise/zh/management-software/smart-provisioning-pid-23143793/software)下载 * Arm服务器：V129及以上版本，可访问[Smart Provisioning](https://support.huawei.com/enterprise/zh/management-software/smart-provisioning-pid-23143793/software)下载 |
| ServiceCD2.0 | V139及以上版本，可访问[FusionServer Tools](http://support.huawei.com/enterprise/zh/server/fusionserver-tools-pid-21015513/software)下载 |

## 安装华为Ansible插件

从[GitHub](https://github.com/Huawei/Server_Management_Plugin_Ansible/tree/master/releases)网站获取华为Ansible插件的安装包。

将安装包上传到Ansible服务器的任意目录（如“/home”目录）。

以root用户登录Ansible服务器。

进入华为Ansible插件的安装包所在目录。

执行以下命令，解压华为Ansible插件的软件包。

**unzip Huawei\_iBMC\_Ansible\_Module\_*x.x*.zip**

执行以下命令，进入解压后的“Huawei\_iBMC\_Ansible\_Module”目录。

**cd Huawei\_iBMC\_Ansible\_Module**

执行以下命令，安装华为Ansible插件。

**python install.py**

安装成功后，在/home目录下会增加一个ibmc\_ansible的文件夹，即为Ansible的安装目录。

----结束

## 卸载华为Ansible插件

以root用户登录Ansible服务器。

进入Ansible的安装目录（如“/home/ibmc\_ansible”），执行卸载命令。

**cd /home/ibmc\_ansible/**

**python uninstall.py**

----结束

# 配置华为Ansible插件

[3.1 配置/etc/ansible/hosts文件](#_ZH-CN_TOPIC_0102786570)

[3.2 配置/group\_vars/myhosts文件](#_ZH-CN_TOPIC_0102786571)

[3.3 配置SSL证书认证和TLS 1.2通讯方式](#_ZH-CN_TOPIC_0214681103)

## 配置/etc/ansible/hosts文件

[myhosts]   
**host0 ibmc\_ip=192.168.2.20 host=huaweiserver0**   
**host1 ibmc\_ip=192.168.2.21 host=huaweiserver1**



第一列的名称（如“host0”、“host1”）不能设置为相同的，否则执行命令时只会针对最后一台服务器执行操作。

## 配置/group\_vars/myhosts文件

设置“/home/ibmc\_ansible/examples/group\_vars”目录下的“myhosts”文件中的相关参数，如iBMC用户名密码、SFTP/CIFS/SCP服务用户名密码、SNMP团体名、OS部署时管理员用户密码等。

[root@localhost examples]# vi /home/ibmc\_ansible/examples/group\_vars/myhosts   
---   
   
# Here we define global variables for our server group, but if some servers   
# require custom values place these variables in /etc/ansible/hosts to override   
# for each individual host   
   
#for create or modify ibmc account   
**account\_user: "account\_user"**   
**account\_pswd: "account\_pswd"**   
   
# input the huawei ibmc user and password   
**ibmc\_user: "ibmc\_user"**   
**ibmc\_pswd: "ibmc\_pwd"**   
   
# input the sftp user and password when we need to use the sftp service   
**sftp\_user: "sftp\_user"**   
**sftp\_pswd: "sftp\_pwd"**   
   
# input the cifs user and password when we need to use the cifs service   
**cifs\_user: "cifs\_user"**   
**cifs\_pswd: "cifs\_pwd"**   
   
# input the scp user and password when we need to use the scp service   
**scp\_user: "scp\_user"**   
**scp\_pswd: "scp\_pwd"**   
   
# if you select SNMP Trap mode as V1 or V2C, you can set the community name   
**community: "community\_name"**   
   
# input the os password when you deploy the server os by sp   
**os\_pswd: "os\_pswd"**



若涉及到密码等敏感数据，建议使用ansible-vault加密方法对“myhosts”文件进行加密，详细请参考<http://docs.ansible.com/ansible/latest/vault.html>。

## 配置SSL证书认证和TLS 1.2通讯方式

参数配置

修改“/home/ibmc\_ansible/examples/set\_request\_cfg.yml”文件。

* TLS 1.2通讯方式通过“force\_tls1\_2”参数进行配置，其为强制使用，即“force\_tls1\_2”参数默认值为“True”。当启用状态下执行命令后出现“import ssl.PROTOCOL\_TLSv1\_2 exception”告警时，需将“force\_tls1\_2”参数设置为“False”。
* SSL证书认证功能通过“verify”和“certify”参数进行配置。若不设置“certify”参数，会使用以下默认的证书库，需先将CA证书导入至对应证书库下。
* 未安装certifi证书库时，默认使用系统的证书库，如：

/etc/pki/tls/certs/ca-bundle.crt

* 安装了certifi证书库时，默认使用此证书库，如：

/usr/lib/python2.7/site-packages/certifi-2019.11.28-py2.7.egg/certifi/cacert.pem

[root@localhost examples]# vi set\_request\_cfg.yml   
   
---   
- hosts: 127.0.0.1   
 connection: local   
 name: set request config   
 gather\_facts: False   
 # verify: the requests module verify server certify or not; Available values: True, False;   
 # certify: the certify use to verify the server, if this params do not set , requests module will used the certificate   
 #which is in the certifi module or the system default certificate. Format: /etc/pki/tls/certs/ca-bundle.crt   
 # force\_tls1\_2: force to use tls1.2 , the default value is true.   
 tasks:   
 - name: set request config   
 ibmc\_set\_redfish\_request\_cfg:   
 **force\_tls1\_2: True**   
 **verify: True**   
 **certify:**

执行配置

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行配置命令。

**ansible-playbook set\_request\_cfg.yml**

如下返回信息表示执行成功。

[root@localhost examples]# ansible-playbook set\_request\_cfg.yml   
   
PLAY [set request config] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [set request config] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [127.0.0.1]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
127.0.0.1 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

# 使用华为Ansible插件

查看帮助信息

1. 执行以下命令查看Ansible插件的所有命令模块名。

**ansible-doc -l |grep ibmc**

[root@localhost ~]# ansible-doc -l |grep ibmc   
[WARNING]: template parsing did not produce documentation.   
[WARNING]: win\_template parsing did not produce documentation.   
ibmc\_ansible\_show\_version Huawei iBMC ansible plugin 2.0.2   
ibmc\_create\_account create a new ibmc account   
ibmc\_create\_raid Create volume   
ibmc\_delete\_account delete bmc account by username   
ibmc\_delete\_raid Delete volume   
ibmc\_deploy\_os\_by\_service\_cd deploy os by service cd   
ibmc\_deploy\_os\_by\_sp deploy os by sp   
ibmc\_get\_account get ibmc accounts info   
ibmc\_get\_basic\_info get ibmc basic info   
ibmc\_get\_boot\_device get boot device   
ibmc\_get\_firmware\_info\_by\_sp get firmware info info   
ibmc\_get\_ip Get ibmc ip info   
ibmc\_get\_ntp Get ntp info   
ibmc\_get\_power\_status get ibmc accounts info   
ibmc\_get\_raid Get raid info   
ibmc\_get\_snmp\_trap Get snmp trap resource info   
ibmc\_inband\_fw\_update update inband firmware   
ibmc\_modify\_account modify ibmc accounts info   
ibmc\_modify\_raid Modify volume   
ibmc\_outband\_fw\_update update outband firmware   
ibmc\_server\_profile export or import the server profile   
ibmc\_set\_boot\_device Set boot device   
ibmc\_set\_ip Set ibmc ip info   
ibmc\_set\_ntp Set ntp info   
ibmc\_set\_power manager server power   
ibmc\_set\_snmp\_trap Set snmp trap info

1. 执行以下命令查看各个命令模块的帮助信息。

**ansible-doc *命令模块名***

如**ansible-doc ibmc\_get\_account**

[root@localhost ~]# ansible-doc ibmc\_get\_account   
> IBMC\_GET\_ACCOUNT (/usr/lib/python2.7/site-packages/ansible-2.8.6-py2.7.egg/ansible/modules/ibmc/ibmc\_get\_account.py)   
   
 get ibmc accounts info   
   
 \* This module is maintained by The Ansible Community   
OPTIONS (= is mandatory):   
   
= ibmc\_ip   
 iBMC IP address   
 [Default: None]   
   
= ibmc\_pswd   
 iBMC user password used for authentication   
 [Default: None]   
   
= ibmc\_user   
 iBMC user name used for authentication   
 [Default: None]   
   
   
 METADATA:   
 status:   
 - preview   
 supported\_by: community   
   
   
EXAMPLES:   
   
- name: get ibmc Account   
 ibmc\_get\_account :   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"

查看插件版本信息

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行以下命令查看版本信息。

**ansible-playbook -v show\_ibmc\_ansible\_version.yml**

[root@localhost examples]# ansible-playbook -v show\_ibmc\_ansible\_version.yml   
   
PLAY [show Huawei iBMC ansible plugin version] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [show Huawei iBMC ansible plugin version] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [127.0.0.1] => {"changed": false, "msg": "**Huawei iBMC ansible modules 2.0.4**"}   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
127.0.0.1 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

[4.1 查询服务器基本信息（生成json文件）](#_ZH-CN_TOPIC_0102786712)

[4.2 管理设备启动项](#_ZH-CN_TOPIC_0206987987)

[4.3 管理电源](#_ZH-CN_TOPIC_0207371483)

[4.4 管理iBMC用户](#_ZH-CN_TOPIC_0193957031)

[4.5 iBMC网络配置](#_ZH-CN_TOPIC_0193957036)

[4.6 管理NTP服务](#_ZH-CN_TOPIC_0193957037)

[4.7 管理SNMP Trap服务](#_ZH-CN_TOPIC_0193957039)

[4.8 导入导出Profile文件](#_ZH-CN_TOPIC_0102786736)

[4.9 固件升级](#_ZH-CN_TOPIC_0102786721)

[4.10 RAID配置](#_ZH-CN_TOPIC_0102786725)

[4.11 OS部署](#_ZH-CN_TOPIC_0102786703)

## 查询服务器基本信息（生成json文件）

功能介绍

支持查询服务器BMC版本、BIOS版本、CPLD版本、序列号、资产标签、服务器型号、服务器健康状态、内存信息和健康状态、CPU信息和健康状态、硬盘信息和健康状态。

操作步骤

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行查询命令。

**ansible-playbook get\_basic\_info.yml**

如下返回信息表示查询成功。

[root@localhost examples]# ansible-playbook get\_basic\_info.yml   
   
PLAY [get bmc basic info] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [get bmc basic info] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

查询信息后生成的json文件（如“172.26.100.9\_BasicInfo.json”）默认保存在“/var/log/ansible/ibmc/report/basic\_info”目录中，建议导出json文件后再查看。

## 管理设备启动项

功能介绍

支持查询和设置启动设备、启动参数的使能状态、启动模式。

### 查询启动项配置信息

操作步骤

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行查询命令。

**ansible-playbook get\_boot\_device.yml**

如下返回信息表示查询成功。

[root@localhost examples]# ansible-playbook get\_boot\_device.yml   
   
PLAY [get boot device] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [get boot device] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

查询结果保存在“/var/log/ansible/ibmc/report”目录下的“ansibleibmc.report”文件中。

1. 执行以下命令查看查询结果。

**cd /var/log/ansible/ibmc/report**

**cat ansibleibmc.report**

[2019-12-02 06:41:10 INFO ] - 172.26.100.10 -- Get boot device info successful! The boot device info is: {'Boot': {u'BootSourceOverrideTarget': u'Hdd', u'BootSourceOverrideMode': u'UEFI', u'BootSourceOverrideEnabled': u'Continuous', u'BootSourceOverrideTarget@Redfish.AllowableValues': [u'None', u'Pxe', u'Floppy', u'Cd', u'Hdd', u'BiosSetup']}}

### 配置启动设备

参数配置

修改“/home/ibmc\_ansible/examples/set\_boot\_device.yml”文件。

[root@localhost examples]# vi set\_boot\_device.yml   
---   
- hosts: myhosts   
 connection: local   
 name: set boot device   
 gather\_facts: False   
   
# boot\_target: Current boot device, Available values: Cd, None, Pxe, Floppy, Hdd, BiosSetup.   
# boot\_enabled: Whether the boot settings are effective, Available values: Disabled, Once, Continuous.   
# boot\_mode: Boot mode, Available values: UEFI, Legacy.   
   
 tasks:   
 - name: set boot device   
 ibmc\_set\_boot\_device:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
  **boot\_target: "Cd"**   
  **boot\_enabled: "Once"**   
 **boot\_mode: "Legacy"**

执行配置

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行配置命令。

**ansible-playbook set\_boot\_device.yml**

如下返回信息表示执行成功。

[root@localhost examples]# ansible-playbook set\_boot\_device.yml   
   
PLAY [set boot device] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [set boot device] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host9]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host9 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

## 管理电源

功能介绍

支持查询和设置电源状态。

### 查询电源状态

操作步骤

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行查询命令。

**ansible-playbook get\_power\_status.yml**

如下返回信息表示查询成功。

[root@localhost examples]# ansible-playbook get\_power\_status.yml   
   
PLAY [get ibmc os power status] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [get ibmc os power status] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

查询结果保存在“/var/log/ansible/ibmc/report”目录下的“ansibleibmc.report”文件中。

1. 执行以下命令查看查询结果。

**cd /var/log/ansible/ibmc/report**

**cat ansibleibmc.report**

[2019-12-02 06:38:32 INFO ] - 172.26.100.10 -- get system power state successful! power status is :Off

### 配置电源状态

参数配置

修改“/home/ibmc\_ansible/examples/set\_power.yml”文件。

[root@localhost examples]# vi set\_power.yml   
---   
- hosts: myhosts   
 connection: local   
 name: power manager   
 gather\_facts: False   
#power\_cmd: Available values:"poweron" "poweroff" "forcerestart" "gracefulshutdown" "forcepowercycle" "nmi"   
 tasks:   
 - name: power manager   
 ibmc\_set\_power:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **power\_cmd: "poweron"**

执行配置

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行配置命令。

**ansible-playbook set\_power.yml**

如下返回信息表示执行成功。

[root@localhost examples]# ansible-playbook set\_power.yml   
   
PLAY [power manager] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [power manager] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host9]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host9 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

## 管理iBMC用户

功能介绍

支持查询、创建、修改和删除iBMC用户。

### 查询iBMC用户（生成json文件）

操作步骤

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行查询命令。

**ansible-playbook get\_account.yml**

如下返回信息表示查询成功。

[root@localhost examples]# ansible-playbook get\_account.yml   
   
PLAY [get ibmc Account] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [get ibmc Account] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

查询信息后生成的json文件（如“172.26.100.9\_AccountInfo.json”）默认保存在“/var/log/ansible/ibmc/report/account\_info”目录中，建议导出json文件后再查看。

### 创建iBMC用户

参数配置

* 修改“/home/ibmc\_ansible/examples/group\_vars/myhosts”文件中的“account\_user”（新建用户名）和“account\_pswd”（新建用户的密码）参数。

[root@localhost examples]# vi /home/ibmc\_ansible/examples/group\_vars/myhosts   
---   
   
# Here we define global variables for our server group, but if some servers   
# require custom values place these variables in /etc/ansible/hosts to override   
# for each individual host   
   
#for create or modify ibmc account   
**account\_user: "account\_user"**   
**account\_pswd: "account\_pswd"**   
   
# input the huawei ibmc user and password   
ibmc\_user: "ibmc\_user"   
ibmc\_pswd: "ibmc\_pwd"   
   
# input the sftp user and password when we need to use the sftp service   
sftp\_user: "sftp\_user"   
sftp\_pswd: "sftp\_pwd"   
   
# input the cifs user and password when we need to use the cifs service   
cifs\_user: "cifs\_user"   
cifs\_pswd: "cifs\_pwd"   
   
# input the scp user and password when we need to use the scp service   
scp\_user: "scp\_user"   
scp\_pswd: "scp\_pwd"   
   
# if you select SNMP Trap mode as V1 or V2C, you can set the community name   
community: "community\_name"   
   
# input the os password when you deploy the server os by sp   
os\_pswd: "os\_pswd"

* 修改“/home/ibmc\_ansible/examples/create\_account.yml”文件。

[root@localhost examples]# vi create\_account.yml   
---   
- hosts: myhosts   
 connection: local   
 name: create ibmc Account   
 gather\_facts: False   
#roleid: role id; Available values: Administrator, Operator, Commonuser, Noaccess, CustomRole1, CustomRole2, CustomRole3, CustomRole4   
 tasks:   
 - name: create ibmc Account   
 ibmc\_create\_account :   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 new\_account\_user: "{{ account\_user }}"   
 new\_account\_pswd: "{{ account\_pswd }}"   
  **roleid: "Administrator"**

执行操作

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行配置命令。

**ansible-playbook create\_account.yml**

如下返回信息表示执行成功。

[root@localhost examples]# ansible-playbook create\_account.yml   
   
PLAY [create ibmc Account] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [create ibmc Account] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host9]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host9 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

### 修改iBMC用户

参数配置

* 修改“/home/ibmc\_ansible/examples/group\_vars/myhosts”文件中的“account\_user”（修改后的用户名）和“account\_pswd”（修改后的用户密码）参数。

[root@localhost examples]# vi /home/ibmc\_ansible/examples/group\_vars/myhosts   
---   
   
# Here we define global variables for our server group, but if some servers   
# require custom values place these variables in /etc/ansible/hosts to override   
# for each individual host   
   
#for create or modify ibmc account   
**account\_user: "account\_user"**   
**account\_pswd: "account\_pswd"**   
   
# input the huawei ibmc user and password   
ibmc\_user: "ibmc\_user"   
ibmc\_pswd: "ibmc\_pwd"   
   
# input the sftp user and password when we need to use the sftp service   
sftp\_user: "sftp\_user"   
sftp\_pswd: "sftp\_pwd"   
   
# input the cifs user and password when we need to use the cifs service   
cifs\_user: "cifs\_user"   
cifs\_pswd: "cifs\_pwd"   
   
# input the scp user and password when we need to use the scp service   
scp\_user: "scp\_user"   
scp\_pswd: "scp\_pwd"   
   
# if you select SNMP Trap mode as V1 or V2C, you can set the community name   
community: "community\_name"   
   
# input the os password when you deploy the server os by sp   
os\_pswd: "os\_pswd"

* 修改“/home/ibmc\_ansible/examples/modify\_account.yml”文件。

[root@localhost examples]# vi modify\_account.yml   
   
- hosts: myhosts   
 connection: local   
 name: modify ibmc Account   
 gather\_facts: False   
#roleid: role id; Available values: Administrator, Operator, Commonuser, Noaccess, CustomRole1, CustomRole2, CustomRole3, CustomRole4   
#locked: it must be False   
#enable: Whether the user is enabled; Available values: True, False   
#login\_interface:list of service the account can access,can be set to empty list []; Available values in list:Web, SNMP, IPMI, SSH, SFTP, Local, Redfish   
#login\_rule: list of login rules,can be set to empty list []; Available values in list:Rule1, Rule2, Rule3   
#account\_insecure\_prompt\_enabled: enable or disable account insecure prompt; Available values: True, False   
 tasks:   
 - name: modify ibmc Account   
 ibmc\_modify\_account :   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 old\_account\_user: "test"   
 new\_account\_user: "{{ account\_user }}"   
 new\_account\_pswd: "{{ account\_pswd }}"   
 **roleid: "Administrator"**   
 **locked: False**   
 **enable: True**   
 **login\_interface:**   
  **- Web**   
 **login\_rule:**   
 **- Rule1**   
 **account\_insecure\_prompt\_enabled: True**

执行操作

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行配置命令。

**ansible-playbook modify\_account.yml**

如下返回信息表示执行成功。

[root@localhost examples]# ansible-playbook modify\_account.yml   
   
PLAY [modify ibmc Account] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [modify ibmc Account] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host9]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host9 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

### 删除iBMC用户

参数配置

修改“/home/ibmc\_ansible/examples/delete\_account.yml”文件中的“delete\_account”（需删除的用户名）参数。

[root@localhost examples]# vi delete\_account.yml   
---   
- hosts: myhosts   
 connection: local   
 name: delete ibmc Account   
 gather\_facts: False   
   
 tasks:   
 - name: delete ibmc Account   
 ibmc\_delete\_account:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **delete\_account: "test"**

执行操作

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行配置命令。

**ansible-playbook delete\_account.yml**

如下返回信息表示执行成功。

[root@localhost examples]# ansible-playbook delete\_account.yml   
   
PLAY [delete ibmc Account] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [delete ibmc Account] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host9]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host9 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

## iBMC网络配置

功能介绍

* 支持查询iBMC网络信息。
* 支持修改IP使能模式、IPv4和IPv6地址信息。

### 查询iBMC网络信息（生成json文件）

操作步骤

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行查询命令。

**ansible-playbook get\_ibmc\_ip.yml**

如下返回信息表示查询成功。

[root@localhost examples]# ansible-playbook get\_ibmc\_ip.yml   
   
PLAY [get ibmc ip] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [get ibmc ip] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

查询信息后生成的json文件（如“172.26.100.9\_iBMCIPInfo.json”）默认保存在“/var/log/ansible/ibmc/report/ibmc\_ip”目录中，建议导出json文件后再查看。

### 配置iBMC网络信息

参数配置

修改“/home/ibmc\_ansible/examples/set\_ibmc\_ip.yml”文件。

[root@localhost examples]# vi set\_ibmc\_ip.yml   
---   
- hosts: myhosts   
 connection: local   
 name: set ibmc ip   
 gather\_facts: False   
   
# ip\_version: Whether IPv4/IPv6 is enabled, Available values: IPv4, IPv6, IPv4AndIPv6.   
# ipv4\_addr: IPv4 address info.   
 # address: IPv4 address.   
 # subnet\_mask: Subnet mask of the IPv4 address.   
 # gateway: Gateway of the IPv4 address.   
 # address\_origin: How the IPv4 address is allocated. Available values: Static, DHCP.   
# ipv6\_addr: IPv6 address info.   
 # address: IPv6 address.   
 # prefix\_length: Prefix length of the IPv6 address, must be an integer, value range: 0 to 128.   
 # address\_origin: How the IPv6 address is allocated. Available values: Static, DHCPv6.   
# ipv6\_gateway: IPv6 gateway address of the iBMC network port.   
   
 tasks:   
 - name: set ibmc ip   
 ibmc\_set\_ip:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **ip\_version: "IPv4AndIPv6"**   
 **ipv4\_addr:**   
 **- address: "172.26.100.100"**   
  **subnet\_mask: "255.255.0.0"**   
 **gateway: "172.26.0.1"**   
 **address\_origin: "Static"**   
 **ipv6\_addr:**   
  **- address: "fc00:172::100"**   
  **prefix\_length: 10**   
 **address\_origin: "Static"**   
 **ipv6\_gateway: "fc00:172::1"**

执行配置

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行配置命令。

**ansible-playbook set\_ibmc\_ip.yml**

如下返回信息表示执行成功。

[root@localhost examples]# ansible-playbook set\_ibmc\_ip.yml   
   
PLAY [set ibmc ip] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [set ibmc ip] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host9]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host9 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

## 管理NTP服务

功能介绍

* 支持查询NTP服务信息。
* 支持配置NTP服务使能、首选/备选NTP服务器地址、服务器身份认证使能、NTP地址模式（IPv4/IPv6/Static）和最小/最大轮询间隔值。

### 查询NTP服务信息

操作步骤

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行查询命令。

**ansible-playbook get\_ntp.yml**

如下返回信息表示查询成功。

[root@localhost examples]# ansible-playbook get\_ntp.yml   
   
PLAY [get ntp] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [get ntp] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

查询结果保存在“/var/log/ansible/ibmc/report”目录下的“ansibleibmc.report”文件中。

1. 执行以下命令查看查询结果。

**cd /var/log/ansible/ibmc/report**

**cat ansibleibmc.report**

[2019-12-02 06:42:10 INFO ] - 172.26.100.10 -- Get NTP configuration resource info successful! The NTP configuration resource info is: {'NtpAddressOrigin': u'Static', 'ServiceEnabled': True, 'ServerAuthenticationEnabled': True, 'MinPollingInterval': 3, 'NTPKeyStatus': u'Uploaded', 'AlternateNtpServer': u'', 'PreferredNtpServer': u'172.26.207.1', 'MaxPollingInterval': 17}

### 配置NTP

参数配置

修改“/home/ibmc\_ansible/examples/set\_ntp.yml”文件。

[root@localhost examples]# vi set\_ntp.yml   
---   
- hosts: myhosts   
 connection: local   
 name: set ntp   
 gather\_facts: False   
   
# service\_enabled: Enable or disable bmc ntp service, Available values: True, False.   
# pre\_ntp\_server: Config preferred NtpServer, you can enter ipv4 ipv6 or domain name, NTP Server will be blanked when set to an empty string.   
# alt\_ntp\_server: Config alternate NtpServer, you can enter ipv4 ipv6 or domain name, NTP Server will be blanked when set to an empty string.   
# server\_auth\_enabled: Enable or disable Server Authentication service, Available values: True, False.   
# ntp\_address\_origin: Config Ntp Address Origin, Available values: IPv4, IPv6, Static.   
# min\_polling\_interval: Config Min Polling Interval time, must be an integer, in 3~17 and <= max\_polling\_interval.   
# max\_polling\_interval: Config Max Polling Interval time, must be an integer, in 3~17 and >= min\_polling\_interval.   
   
 tasks:   
 - name: set ntp   
 ibmc\_set\_ntp:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **service\_enabled: True**   
 **pre\_ntp\_server: "192.168.2.10"**   
 **alt\_ntp\_server: "192.168.2.20**"   
 **server\_auth\_enabled: False**   
 **ntp\_address\_origin: "Static"**   
 **min\_polling\_interval: 3**   
 **max\_polling\_interval: 17**

执行配置

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行配置命令。

**ansible-playbook set\_ntp.yml**

如下返回信息表示执行成功。

[root@localhost examples]# ansible-playbook set\_ntp.yml   
   
PLAY [set ntp] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [set ntp] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host9]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host9 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

## 管理SNMP Trap服务

功能介绍

* 支持查询SNMP的Trap服务。
* 支持配置Trap功能的使能状态、TrapV3用户名、上报模式、主机标识、团体名、告警发送级别和Trap服务器。

### 查询SNMP服务信息（生成json文件）

操作步骤

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行查询命令。

**ansible-playbook get\_snmp\_trap.yml**

如下返回信息表示查询成功。

[root@localhost examples]# ansible-playbook get\_snmp\_trap.yml   
   
PLAY [get snmp trap] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [get snmp trap] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

查询信息后生成的json文件（如“172.26.100.9\_SNMPTrapInfo.json”）默认保存在“/var/log/ansible/ibmc/report/snmp\_trap”目录中，建议导出json文件后再查看。

### 配置SNMP Trap

参数配置

* 修改“/home/ibmc\_ansible/examples/set\_snmp\_trap.yml”文件。

[root@localhost examples]# vi set\_snmp\_trap.yml   
---   
- hosts: myhosts   
 connection: local   
 name: set snmp trap   
 gather\_facts: False   
   
# service\_enabled: Whether trap is enabled, Available values: True, False.   
# trap\_version: Trap version, Available values: V1, V2C, V3.   
# trap\_v3\_user: SNMPv3 user name, valid only for trap version is V3.   
# trap\_mode: Trap mode, Available values: OID, EventCode, PreciseAlarm.   
# trap\_server\_identity: Host identifier, Available values: BoardSN, ProductAssetTag, HostName.   
# alarm\_severity: Severity levels of the alarm to be sent, Available values: Critical, Major, Minor, Normal.   
# trap\_servers: Can set one or more trap server, When all parameters of the trap server are empty, it indicates that the trap server is not configured.   
 # trap\_server\_enabled: Whether the trap server is enabled, Available values: True, False.   
 # trap\_server\_address: Server address, you can enter ipv4 ipv6 or domain name.   
 # trap\_server\_port: Server port number, must be an integer, Available value range: 1 to 65535.   
   
 tasks:   
 - name: set snmp trap   
 ibmc\_set\_snmp\_trap:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 community: "{{ community }}"   
 **service\_enabled: True**   
 **trap\_version: "V3"**   
 **trap\_v3\_user: "root"**   
  **trap\_mode: "OID"**   
 **trap\_server\_identity: "HostName"**   
 **alarm\_severity: "Normal"**   
 **trap\_servers:**   
 **- trap\_server\_enabled: True**   
 **trap\_server\_address: "192.168.2.10"**   
 **trap\_server\_port: 160**   
  **- trap\_server\_enabled: True**   
 **trap\_server\_address: "192.168.2.11"**   
  **trap\_server\_port: 161**   
 **- trap\_server\_enabled: False**   
  **trap\_server\_address: "192.168.2.12"**   
  **trap\_server\_port: 162**   
 - **trap\_server\_enabled: False**   
 **trap\_server\_address: "192.168.2.13"**   
 **trap\_server\_port: 163**

* 修改“/home/ibmc\_ansible/examples/group\_vars/myhosts”文件中的“community”（团体名称）参数。



当“trap\_version”配置为“V1”或者“V2C”时，可在“/home/ibmc\_ansible/examples/group\_vars/myhosts”文件中配置“community”参数。

[root@localhost examples]# vi /home/ibmc\_ansible/examples/group\_vars/myhosts   
---   
   
# Here we define global variables for our server group, but if some servers   
# require custom values place these variables in /etc/ansible/hosts to override   
# for each individual host   
   
#for create or modify ibmc account   
account\_user: "account\_user"   
account\_pswd: "account\_pswd"   
   
# input the huawei ibmc user and password   
ibmc\_user: "ibmc\_user"   
ibmc\_pswd: "ibmc\_pwd"   
   
# input the sftp user and password when we need to use the sftp service   
sftp\_user: "sftp\_user"   
sftp\_pswd: "sftp\_pwd"   
   
# input the cifs user and password when we need to use the cifs service   
cifs\_user: "cifs\_user"   
cifs\_pswd: "cifs\_pwd"   
   
# input the scp user and password when we need to use the scp service   
scp\_user: "scp\_user"   
scp\_pswd: "scp\_pwd"   
   
# if you select SNMP Trap mode as V1 or V2C, you can set the community name   
**community: "community\_name"**   
   
# input the os password when you deploy the server os by sp   
os\_pswd: "os\_pswd"

执行配置

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行配置命令。

**ansible-playbook set\_snmp\_trap.yml**

如下返回信息表示执行成功。

[root@localhost examples]# ansible-playbook set\_snmp\_trap.yml   
   
PLAY [set snmp trap] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [set snmp trap] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host9]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host9 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

## 导入导出Profile文件

功能介绍

本功能实现了BIOS和iBMC的Profile配置文件导入和导出功能，支持本地或远程的方式将Profile文件导入到服务器，或者支持本地或远程的方式将服务器的Profile文件导出到本地iBMC的“/tmp”目录下。

参数配置

修改“/home/ibmc\_ansible/examples/server\_profile.yml”文件。



使用iBMC本地临时目录时，必须为“/tmp”目录。

[root@localhost examples]# vi server\_profile.yml   
---   
- hosts: myhosts   
 connection: local   
 name: server profile   
 gather\_facts: False   
#file\_name: the file name you want to import or export ;if the file name is empty ,ibmc\_ansible\_profile will used the default name; such as 172.26.201.2\_profile.xml   
#command: Available values:export, import   
#file\_path: Local export: /tmp Remote export: File protocol://Username:Password@IPaddress/Folder The file transfer   
# protocols: Available values: sftp,https,nfs,cifs,scp   
   
 tasks:   
 - name: server profile   
 ibmc\_server\_profile:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **command: "export"**   
 **file\_path: "/tmp"**   
 **file\_name: "profile.xml"**

执行操作

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行配置命令。

**ansible-playbook server\_profile.yml**

如下返回信息表示执行成功。

[root@localhost examples]# ansible-playbook server\_profile.yml   
   
PLAY [server profile] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [server profile] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host9]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host9 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

## 固件升级

功能介绍

* 支持查询固件版本信息。
* 支持升级带外固件，包括BMC、BIOS、CPLD。
* 支持Smart Provisioning方式的带内固件升级。

### 查询固件版本信息（生成json文件）



在使用其他工具升级带内固件后，需要重新启动Smart Provisioning，才能获取最新的固件版本信息。

操作步骤

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行查询命令。

**ansible-playbook get\_firmware\_info\_by\_sp.yml**

如下返回信息表示查询成功。

[root@localhost examples]# ansible-playbook get\_firmware\_info\_by\_sp.yml   
   
PLAY [get firmware info by sp] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [get firmware info by sp] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host9]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host9 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

查询信息后生成的json文件（如“172.26.100.9\_fwInfo.json”）默认保存在“/var/log/ansible/ibmc/report/inband\_fw\_info”目录中，建议导出json文件后再查看。

### 升级固件

#### 带外固件升级

参数配置

修改“/home/ibmc\_ansible/examples/update\_outband\_fw.yml”文件中的“image\_url”和“protocol”参数。

* image\_url：升级文件的路径，输入形式为“文件传输协议://用户名:密码@服务器IP地址/目录/文件名”。
* protocol：文件传输协议，包括SFTP、HTTPS、NFS、CIFS、SCP。



升级前请确认固件包中的hpm文件已上传至文件服务器相应路径下。固件包的获取方法如下：

登录[技术支持 > 智能服务器](https://support.huawei.com/enterprise/zh/category/intelligent-servers-pid-1548148142425?submodel=15791)。

选择对应的服务器型号进入目录。

选择“软件”页签。

选择补丁版本路径。

下载所需的固件包。

[root@localhost examples]# vi update\_outband\_fw.yml   
---   
- hosts: myhosts   
 connection: local   
 name: update outband fw   
 gather\_facts: False   
   
 tasks:   
 - name: update outband fw   
 ibmc\_outband\_fw\_update:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **image\_url: "nfs://172.26.200.11/tmp/package/cpldimage.hpm"**   
 **protocol: "NFS**"

执行升级

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行配置命令。

**ansible-playbook update\_outband\_fw.yml**

如下返回信息表示执行成功。

[root@localhost examples]# ansible-playbook update\_outband\_fw.yml   
   
PLAY [update outband fw] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [update outband fw] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host9]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host9 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

#### 带内固件升级

功能介绍

支持Smart Provisioning方式的带内固件升级。

具体支持的服务器以及具体支持的固件升级信息请参见以下对应文档：

* x86服务器：[《Smart Provisioning 用户指南》](https://support.huawei.com/enterprise/zh/doc/EDOC1100034051)
* Arm服务器：[《Smart Provisioning 用户指南 (aarch64)》](https://support.huawei.com/enterprise/zh/doc/EDOC1100108319)

参数配置

修改“/home/ibmc\_ansible/examples/update\_inband\_fw.yml”文件中的“image\_url”参数。

image\_url：升级文件的路径，输入形式为“文件传输协议://用户名:密码@服务器IP地址/目录/文件名”。文件传输协议包括SFTP、HTTPS、NFS、CIFS、SCP。



升级前请确认固件升级包和数字签名文件已上传至文件服务器相应路径下。固件升级包和数字签名文件请访问[FusionServer iDriver](https://support.huawei.com/enterprise/zh/management-software/fusionserver-idriver-pid-21588909/software)获取。

[root@localhost examples]# vi update\_inband\_fw.yml   
---   
- hosts: myhosts   
 connection: local   
 name: update inband fw   
 gather\_facts: False   
   
 tasks:   
 - name: update inband fw   
 ibmc\_inband\_fw\_update:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **image\_url:**   
 **- "sftp://172.26.200.11/data/NIC-LOM-X722-10GE\_SFP-GE\_Electrical-FW-3.33\_0x80000f09.zip"**   
 file\_server\_user: "{{sftp\_user}}"   
 file\_server\_pswd: "{{sftp\_pswd}}"

文件服务器的用户名和密码需在“/home/ibmc\_ansible/examples/group\_vars”目录下的“myhosts”文件中修改。

[root@localhost examples]# vi /home/ibmc\_ansible/examples/group\_vars/myhosts   
---   
   
# Here we define global variables for our server group, but if some servers   
# require custom values place these variables in /etc/ansible/hosts to override   
# for each individual host   
   
#for create or modify ibmc account   
account\_user: "account\_user"   
account\_pswd: "account\_pswd"   
   
# input the huawei ibmc user and password   
ibmc\_user: "ibmc\_user"   
ibmc\_pswd: "ibmc\_pwd"   
   
# input the sftp user and password when we need to use the sftp service   
**sftp\_user: "sftp\_user"**   
**sftp\_pswd: "sftp\_pwd"**   
   
# input the cifs user and password when we need to use the cifs service   
**cifs\_user: "cifs\_user"**   
**cifs\_pswd: "cifs\_pwd"**   
   
# input the scp user and password when we need to use the scp service   
**scp\_user: "scp\_user"**   
**scp\_pswd: "scp\_pwd"**   
   
# if you select SNMP Trap mode as V1 or V2C, you can set the community name   
community: "community\_name"   
   
# input the os password when you deploy the server os by sp   
os\_pswd: "os\_pswd"

执行升级

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行配置命令。

**ansible-playbook update\_inband\_fw.yml**

如下返回信息表示执行成功。

[root@localhost examples]# ansible-playbook update\_inband\_fw.yml   
   
PLAY [update inband fw] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [update inband fw] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

## RAID配置

功能介绍

* 仅支持配置带外管理的RAID卡。
* 支持多RAID卡的场景。
* 支持RAID卡（LSI SAS3108、Avago SAS3408iMR、Avago SAS3004iMR、Avago SAS3508）的查询、配置、修改和删除。

### 查询RAID配置（生成json文件）

操作步骤

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行查询命令。

**ansible-playbook get\_raid.yml**

如下返回信息表示查询成功。

[root@localhost examples]# ansible-playbook get\_raid.yml   
   
PLAY [get raid] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [get raid] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

查询信息后生成的json文件（如“172.26.100.9\_RAIDInfo.json”）默认保存在“/var/log/ansible/ibmc/report/raid”目录中，建议导出json文件后再查看。

### 删除RAID组

参数配置

修改“/home/ibmc\_ansible/examples/delete\_raid.yml”文件。

[root@localhost examples]# vi delete\_raid.yml   
---   
- hosts: myhosts   
 connection: local   
 name: delete raid   
 gather\_facts: False   
   
# storage\_id: ID of the storage resource   
 # 1.Delete one RAID storage, Format: RAIDStorage+Controller\_ID   
 # 2.Delete multiple RAID storage, Separated by commas, Format: RAIDStorage+Controller\_ID1,RAIDStorage+Controller\_ID2,...   
 # 3.Delete all RAID storage, Format: all   
# volume\_id: Volume resource ID   
 # 1.Delete one volume, Format: LogicalDrive+Volume\_ID   
 # 2.Delete multiple volume, Separated by commas, Format: LogicalDrive+Volume\_ID1,LogicalDrive+Volume\_ID2,...   
 # 3.Delete all volume, Format: all   
   
 tasks:   
 - name: delete raid   
 ibmc\_delete\_raid:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **storage\_id: "RAIDStorage0,RAIDStorage1"**   
 **volume\_id: "LogicalDrive0,LogicalDrive1"**

执行配置

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行配置命令。

**ansible-playbook delete\_raid.yml**

如下返回信息表示执行成功。

[root@localhost examples]# ansible-playbook delete\_raid.yml   
   
PLAY [delete raid] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [delete raid] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host9]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host9 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

### 创建RAID组

参数配置

修改“/home/ibmc\_ansible/examples/create\_raid.yml”文件。



不同的RAID卡支持可配置的参数不同，具体请参考[《华为V2&V3服务器 RAID 控制卡 用户指南》](https://support.huawei.com/enterprise/zh/doc/EDOC1000004345)、[《华为V5服务器 RAID控制卡 用户指南》](https://support.huawei.com/enterprise/zh/doc/EDOC1000163568?idPath=7919749%7C9856522%7C21782478%7C21872244)或[《TaiShan 服务器 RAID控制卡 用户指南》](https://support.huawei.com/enterprise/zh/doc/EDOC1100048779)。

[root@localhost examples]# vi create\_raid.yml   
---   
- hosts: myhosts   
 connection: local   
 name: create raid   
 gather\_facts: False   
   
# storage\_id: ID of the storage resource. Format: RAIDStorage+Controller\_ID   
# capacity\_mbyte: Volume capacity, must be an integer, the size unit is MB. It is an optional parameter   
# stripe\_size: Stripe size of a volume, must be an integer. It is an optional parameter. Available values: 65536, 131072, 262144, 524288, 1048576   
# cachecade\_flag: Whether it is a CacheCade volume. It is an optional parameter, Available values: True, False   
# drives: Member disk list. It is a mandatory parameter. Format: "disk1,disk2,.,diskN"   
# volume\_raid\_level: RAID level of the volume. It is a mandatory parameter. Available values: RAID0, RAID1, RAID5, RAID6, RAID10, RAID50, RAID60   
# volume\_name: Volume name. It is an optional parameter. A string of up to 15 bytes. Value range: ASCII code corresponding to 0x20 to 0x7E   
# df\_read\_policy: Default read policy of the volume. It is an optional parameter. Available values: NoReadAhead, ReadAhead   
# df\_write\_policy: Default write policy of the volume. It is an optional parameter. Available values: WriteThrough, WriteBackWithBBU, WriteBack   
# df\_cache\_policy: Default cache policy of the volume. It is an optional parameter. Available values: CachedIO, DirectIO   
# span\_num: Number of spans of the volume, must be an integer. It is an optional parameter   
 # 1.Set this parameter to 1 when creating a RAID0, RAID1, RAID5, or RAID6 array.   
 # 2.Set this parameter to a value from 2 to 8 when creating a RAID10, RAID50, or RAID60 array.   
# access\_policy: Volume access policy. It is an optional parameter. Available values: ReadWrite, ReadOnly, Blocked   
# disk\_cache\_policy: Cache policy for member disks. It is an optional parameter. Available values: Unchanged, Enabled, Disabled   
# init\_mode: Volume initialization mode. It is an optional parameter. Available values: UnInit, QuickInit, FullInit   
   
   
 tasks:   
 - name: create raid   
 ibmc\_create\_raid:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **volumes:**   
 **- storage\_id: "RAIDStorage0"**   
 **capacity\_mbyte: 1000**   
 **stripe\_size: 65536**   
 **cachecade\_flag: False**   
 **drives: "0,1"**   
 **volume\_raid\_level: "RAID0"**   
  **volume\_name: "volume\_name"**   
 **df\_read\_policy: "NoReadAhead"**   
 **df\_write\_policy: "WriteThrough"**   
 **df\_cache\_policy: "CachedIO"**   
 **span\_num: 1**   
 **access\_policy: "ReadWrite"**   
  **disk\_cache\_policy: "Unchanged"**   
 **init\_mode: "UnInit"**

执行配置

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行配置命令。

**ansible-playbook create\_raid.yml**

如下返回信息表示执行成功。

[root@localhost examples]# ansible-playbook create\_raid.yml   
   
PLAY [create raid] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [create raid] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host9]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host9 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

### 修改RAID配置

参数配置

修改“/home/ibmc\_ansible/examples/modify\_raid.yml”文件。



不同的RAID卡支持可修改的参数不同，具体请参考[《华为V2&V3服务器 RAID 控制卡 用户指南》](https://support.huawei.com/enterprise/zh/doc/EDOC1000004345)、[《华为V5服务器 RAID控制卡 用户指南》](https://support.huawei.com/enterprise/zh/doc/EDOC1000163568?idPath=7919749%7C9856522%7C21782478%7C21872244)或[《TaiShan 服务器 RAID控制卡 用户指南》](https://support.huawei.com/enterprise/zh/doc/EDOC1100048779)。

[root@localhost examples]# vi modify\_raid.yml   
---   
- hosts: myhosts   
 connection: local   
 name: modify raid   
 gather\_facts: False   
   
# storage\_id: ID of the storage resource. Format: RAIDStorage+Controller\_ID   
# volume\_id: Volume resource ID. Format: LogicalDrive+Volume\_ID   
# volume\_name: Volume name. It is an optional parameter. A string of up to 15 bytes. Value range: ASCII code corresponding to 0x20 to 0x7E   
# df\_read\_policy: Default read policy of the volume. It is an optional parameter. Available values: NoReadAhead, ReadAhead   
# df\_write\_policy: Default write policy of the volume. It is an optional parameter. Available values: WriteThrough, WriteBackWithBBU, WriteBack   
# df\_cache\_policy: Default cache policy of the volume. It is an optional parameter. Available values: CachedIO, DirectIO   
# boot\_enable: Whether it is the boot device. Available values: True.   
# bgi\_enable: Whether background initialization is enabled. Available values: True, False.   
# access\_policy: Volume access policy. It is an optional parameter. Available values: ReadWrite, ReadOnly, Blocked   
# ssd\_cache\_enable: Whether the CacheCade volume is used as the cache. Available values: True, False.   
# disk\_cache\_policy: Cache policy for member disks. It is an optional parameter. Available values: Unchanged, Enabled, Disabled   
   
   
 tasks:   
 - name: modify raid   
 ibmc\_modify\_raid:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 volumes:   
 - **storage\_id: "RAIDStorage0"**   
 **volume\_id: "LogicalDrive0"**   
 **volume\_name: "volume\_name"**   
 **df\_read\_policy: "NoReadAhead"**   
 **df\_write\_policy: "WriteThrough"**   
 **df\_cache\_policy: "CachedIO"**   
 **boot\_enable: True**   
 **bgi\_enable: False**   
 **access\_policy: "ReadWrite"**   
 **ssd\_cache\_enable: False**   
 **disk\_cache\_policy: "Unchanged"**

执行配置

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行配置命令。

**ansible-playbook modify\_raid.yml**

如下返回信息表示执行成功。

[root@localhost examples]# ansible-playbook modify\_raid.yml   
   
PLAY [modify raid] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [modify raid] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host9]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host9 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

## OS部署



* 待部署OS的服务器需提前完成RAID配置。
* 使用ServiceCD2.0方式时，部署OS的服务器的逻辑盘的容量不能超过2T，否则ServiceCD2.0无法识别。

### ServiceCD2.0方式



* ServiceCD2.0方式仅支持x86 V2/V3服务器。
* ServiceCD2.0方式依赖ServiceCD2.0镜像包，ServiceCD2.0镜像包请访问[FusionServer Tools](https://support.huawei.com/enterprise/zh/management-software/fusionserver-tools-pid-21015513/software/)获取。

参数配置

修改“/home/ibmc\_ansible/examples/deploy\_os\_by\_service\_cd.yml”文件。

关于各参数的具体信息请参见表4-1。

[root@localhost examples]# vi deploy\_os\_by\_service\_cd.yml   
   
---   
- hosts: myhosts   
 connection: local   
 name: deploy os by service cd   
 gather\_facts: False   
#os\_img: The os image path ; Format: protocol://Username:Password@IPaddress/Folder/image\_file; Available protocol: nfs,cifs,https   
#service\_cd\_img:The service cd image path; Format: protocol://Username:Password@IPaddress/Folder/image\_file; Available protocol: nfs,cifs,https   
#os\_Type: os type ; Available values:   
 # CentOS6U7\_x64, CentOS6U8\_x64, CentOS6U9\_x64,CentOS7U0\_x64, CentOS7U1\_x64, CentOS7U2\_x64, CentOS7U3\_x64, CentOS7U4\_x64, CentOS7U5\_x64   
 # RHEL6U7\_x64, RHEL6U8\_x64, RHEL6U9\_x64, RHEL7U0\_x64, RHEL7U1\_x64, RHEL7U2\_x64, RHEL7U3\_x64, RHEL7U4\_x64, RHEL7U5\_x64,   
 # SLES11SP4\_x64, SLES11SP3\_x64, SLES12\_x64, SLES12SP1\_x64, SLES12SP2\_x64, SLES12SP3\_x64,   
 # Ubuntu16.04\_x64, Ubuntu14.04\_x64   
 # ESXi5.5\_x64, ESXi6.0\_x64, ESXi6.5\_x64, ESXi6.7\_x64   
 # Win2008\_R2\_x64, Win2012\_x64, Win2012\_R2\_x64, Win2016\_x64,   
#cd\_key: key of the OS Installation   
#password: Available values: Please refer to the installation guide of the OS for more infomation. Tips: use a strong password otherwise may failed to install os   
#timezone: Available values: windows Format: (GMT-12:00) International Date Line West; centos,redhat,ubuntu Format:Africa/Abidjan;   
 # SLES Format: Africa/Abidjan ; Please refer to the installation guide of the OS for more infomation.   
#language: Available values: windows Format: en-US; centos,redhat,ubuntu Format:en\_US.UTF-8; SlES Format:en\_US   
 # Please refer to the installation guide of the OS for more infomation.   
#hostname: Host Name   
#owner\_name: Owner Name   
#org\_name: Organize Name   
#position: Position where the os install; Available values: disk, usb   
#partitions: Partition information; Available values: Please refer to the installation guide of the OS.   
#mode: mode to install Available values: 1, 2, 3, ;1 for standard, 2 for full , 3 for Customized   
#rpms: rpm packages you want to install.   
#script: install script   
#software: software you want to install; Available values: ibma   
#win\_os\_name: windows os name, only for windows. Available values:   
 # Windows Server 2016 ServerStandard, Windows Server 2016 ServerStandardCore, Windows Server 2016 ServerDataCenter, Windows Server 2016 ServerDataCenterCore   
 # Windows Server 2012 R2 ServerStandard, Windows Server 2012 R2 ServerStandardCore, Windows Server 2012 R2 ServerDataCenter, Windows Server 2012 R2 ServerDataCenterCore   
 # Windows Server 2012 ServerStandard, Windows Server 2012 ServerStandardCore, Windows Server 2012 ServerDataCenter,Windows Server 2012 ServerDataCenterCore   
 # Windows Server 2008 R2 ServerStandard, Windows Server 2008 R2 ServerStandardCore, Windows Server 2008 R2 ServerEnterprise,   
 # Windows Server 2008 R2 ServerEnterpriseCore, Windows Server 2008 R2 ServerDataCenter, Windows Server 2008 R2 ServerDataCenterCore   
 # Windows Server 2008 R2 ServerWeb,Windows Server 2008 R2 ServerWebCore   
 tasks:   
 - name: deploy os by service cd   
 ibmc\_deploy\_os\_by\_service\_cd:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **service\_cd\_img: "nfs://172.26.200.11/data/serviceCD.iso"**   
 **os\_img: "nfs://172.26.200.11/data/CentOS-7.3-x86\_64-DVD-1611.iso"**   
 **os\_type: "CentOS7U3\_x64"**   
 **win\_os\_name:**   
 **cd\_key:**   
 **password: "{{ os\_pswd }}"**   
 **hostname:**   
 **owner\_name:**   
 **language: "en\_US.UTF-8"**   
 **org\_name:**   
 **position: "disk"**   
 **partitions:**   
  **- partition: "swap:swap:10000|/:ext3:1"**   
 **timezone: "America/New\_York"**   
 **mode:**   
 **rpms:**   
 **- rpm:**   
  **script:**   
 **software: "ibma"**

参数说明

| 参数名称 | 参数说明 | 取值 |
| --- | --- | --- |
| service\_cd\_img | ServiceCD2.0镜像路径（必填） | 输入形式：文件传输协议://用户名:密码@服务器IP地址/目录/文件名  文件传输协议包括NFS、CIFS、HTTPS |
| os\_img | OS镜像路径（必填） |
| os\_type | 安装OS类型（必填） | 可取值包括：  CentOS6U7\_x64、CentOS6U8\_x64、CentOS6U9\_x64、CentOS7U0\_x64、CentOS7U1\_x64、CentOS7U2\_x64、CentOS7U3\_x64、CentOS7U4\_x64、CentOS7U5\_x64  RHEL6U7\_x64、RHEL6U8\_x64、RHEL6U9\_x64、RHEL7U0\_x64、RHEL7U1\_x64、RHEL7U2\_x64、RHEL7U3\_x64、RHEL7U4\_x64、RHEL7U5\_x64  SLES11SP4\_x64、SLES11SP3\_x64、SLES12\_x64、SLES12SP1\_x64、SLES12SP2\_x64、SLES12SP3\_x64  Ubuntu16.04\_x64、Ubuntu14.04\_x64  ESXi5.5\_x64、ESXi6.0\_x64、ESXi6.5\_x64、ESXi6.7\_x64  Win2008\_R2\_x64、Win2012\_x64、Win2012\_R2\_x64、Win2016\_x64 |
| win\_os\_name | Windows系统名称（必填） | 仅支持Windows系统，可取值：  Windows Server 2016 ServerStandard、Windows Server 2016 ServerStandardCore、Windows Server 2016 ServerDataCenter、Windows Server 2016 ServerDataCenterCore  Windows Server 2012 R2 ServerStandard、Windows Server 2012 R2 ServerStandardCore、Windows Server 2012 R2 ServerDataCenter、Windows Server 2012 R2 ServerDataCenterCore  Windows Server 2012 ServerStandard、Windows Server 2012 ServerStandardCore、Windows Server 2012 ServerDataCenter、Windows Server 2012 ServerDataCenterCore  Windows Server 2008 R2 ServerStandard、Windows Server 2008 R2 ServerStandardCore、Windows Server 2008 R2 ServerEnterprise、Windows Server 2008 R2 ServerEnterpriseCore、Windows Server 2008 R2 ServerDataCenter、Windows Server 2008 R2 ServerDataCenterCore、Windows Server 2008 R2 ServerWeb、Windows Server 2008 R2 ServerWebCore  其他系统为空 |
| cd\_key | OS系统的安装秘钥（选填） | Windows和Vmware：选填 |
| password | 管理员初始密码（必填） | 在myhosts文件中设置该参数：os\_pswd |
| hostname | 主机名称（选填） | - |
| owner\_name | 所有者名称（选填） | - |
| language | 系统语言（选填） | 取值请参考对应系统的安装指导文档   * Linux：必填（例：RHEL/CentOS/Ubuntu:"en\_US.UTF-8"；SLES:"en\_US"） * Windows：必填（例："en-US"） * Vmware：为空 |
| org\_name | 组织名称（选填） | - |
| position | 系统部署位置（选填） | * disk * usb |
| partitions | 系统分区设置（选填） | 取值请参考对应系统的安装指导文档   * Linux：必填（例："swap:swap:10000|/:ext3:1"） * Windows：必填（例："C:NTFS:50000|D:NTFS:1"） * Vmware：为空 |
| timezone | 系统时区（选填） | 取值请参考对应系统的安装指导文档   * Linux：必填（例："America/New\_York"） * Windows：必填（例："(GMT-12:00) International Date Line West"） * Vmware：为空 |
| mode | 系统安装模式（选填） | * 1 * 2 * 3   1表示Standard；2表示full；3表示Customized |
| rpms | 系统需安装的rpm包（选填） | - |
| script | 安装脚本（选填） | - |
| software | 系统需安装的软件（选填） | ibma |

执行配置

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行配置命令。

**ansible-playbook deploy\_os\_by\_service\_cd.yml**

如下返回信息表示执行成功。

[root@localhost examples]# ansible-playbook deploy\_os\_by\_service\_cd.yml   
   
PLAY [deploy os by service cd] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [deploy os by service cd] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

### Smart Provisioning方式

Smart Provisioning方式具体支持的服务器请参见以下对应文档：

* x86服务器：[《Smart Provisioning 用户指南》](https://support.huawei.com/enterprise/zh/doc/EDOC1100034051)
* Arm服务器：[《Smart Provisioning 用户指南 (aarch64)》](https://support.huawei.com/enterprise/zh/doc/EDOC1100108319)



此方式包括“deploy\_centos7u3\_by\_sp.yml”、“deploy\_esxi65\_by\_sp.yml”和“deploy\_win2012r2\_by\_sp.yml”命令，以下以“deploy\_centos7u3\_by\_sp.yml”为例进行说明。

参数配置

修改“/home/ibmc\_ansible/examples/deploy\_centos7u3\_by\_sp.yml”文件。

关于各参数的具体信息请参见表4-2。

[root@localhost examples]# vi deploy\_centos7u3\_by\_sp.yml   
   
---   
- hosts: myhosts   
 connection: local   
 name: ibmc deploy centos7u3 by sp   
 gather\_facts: False   
#os\_img: The os image path ; Format: protocol://Username:Password@IPaddress/Folder/image\_file; Available protocol: nfs,cifs,https   
#OSType:Os type; Available values:RHEL6U9, RHEL6U10, RHEL7U3 ,RHEL7U4, RHEL7U5, RHEL7U6, CentOS6U9, CentOS6U10, CentOS7U3, CentOS7U4, CentOS7U5, CentOS7U6, ESXi6.0, ESXi6.5, ESXi6.7   
# SLES11SP4, SLES12SP2, SLES12SP3, Ubuntu16.04, Ubuntu16.04.1, Ubuntu16.04.2, Win2016, Win2016 Standard Desktop, Win2016 Standard Core, Win2016 Datacenter Desktop   
# Win2016 Datacenter Core, Win2012\_R2, Win2012\_R2 Standard Desktop, Win2012\_R2 Standard Core, Win2012\_R2 Datacenter Desktop, Win2012\_R2 Datacenter Core   
# EulerOSV2SP3   
#InstallMode: OS Installation mode; Available values:Recommended, Customized   
#Language: Available values: Please refer to the installation guide of the OS.   
#TimeZone: Available values: Please refer to the installation guide of the OS.   
#Keyboard: Available values: Please refer to the installation guide of the OS.   
#BootType: Bios boot mode,This parameter is optional; Available values: UEFIBoot, LegacyBoot, SecureBoot   
#CDKey: key of the OS Installation, This parameter is optional   
#RootPwd: Root user password, this parameter is mandatory;   
# Windows: a sting of at least 6 characters.SUSE: a sting of at least 6 characters.   
# Centos/Redhat/ EulerOS: a sting of at least 6 characters excluding #,$, and space.   
# Ubuntu: a sting of at least 8 characters excluding #, $, and space.   
# Vmware: a string of 7 to 40 characters. For the ESXi 6.7, the password must consist of letters, digits, and special characters.   
# (NOTE: Smart Provisioning supports special characters #, $ and spaces from V119.)   
#HostName: Host Name,This parameter is optional Installation   
#CheckFirmware: Whether to verify firmware.This parameter is optional; Available values:True, False   
#Partition: Partition information. This parameter is optional. Please refer to the installation guide of the OS   
#Software: Software list. This parameter is mandatory.Format:{ "FileName": "iBMA" }   
#Autopart: Whether auto-partitioning is supported. Linux/VMware: true Window: false   
#MediaType:Type of the media where the OS can be deployed.This parameter is optional; Available values:SANBoot, Disk, USB   
#AutoPosition: Whether the installation drive is automatically selected ; Available values:True   
#NetCfg: Network config   
 tasks:   
 - name: ibmc deploy centos7u3 by sp   
 ibmc\_deploy\_os\_by\_sp:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **os\_img: "nfs://172.26.200.11/data/centeros7u3.iso"**   
 **os\_config:**   
 **InstallMode: "Recommended"**   
 **OSType: "CentOS7U3"**   
 **BootType: "UEFIBoot"**   
 **CDKey: ""**   
 **RootPwd: "{{ os\_pswd }}"**   
 **HostName: "test"**   
 **Language: "en\_US.UTF-8"**   
 **TimeZone: "America/New\_York"**   
 **Keyboard: "us"**   
 **CheckFirmware: False**   
 **Partition: []**   
 **Autopart: True**   
 **AutoPosition: True**   
 **Software: []**   
 **NetCfg:**   
 **- Device:**   
 **Name: "eth10086"**   
 **MAC: "04:B0:E7:48:27:84"**   
 **IPv4Addresses:**   
 **- Address: "192.168.2.44"**   
 **SubnetMask: "255.255.0.0"**   
 **Gateway: "192.168.2.1"**   
 **AddressOrigin: "Static"**   
 **IPv6Addresses:**   
 **- Address: ""**   
 **PrefixLength: ""**   
 **Gateway: ""**   
 **AddressOrigin: "Static"**   
 **NameServers:**   
 **- DNS: "192.168.2.1"**   
 **- DNS: "192.168.2.2"**

参数说明

| 参数名称 | 参数说明 | 取值 |
| --- | --- | --- |
| os\_img | OS镜像路径（必填） | 输入形式：文件传输协议://用户名:密码@服务器IP地址/目录/文件名  文件传输协议包括NFS、CIFS、HTTPS |
| InstallMode | 安装模式（必填） | * Recommended * Customized |
| OSType | 安装OS类型（必填） | 可取值包括：  RHEL6U9、RHEL6U10、RHEL7U3、RHEL7U4、RHEL7U5、RHEL7U6、CentOS6U9、CentOS6U10、CentOS7U3、CentOS7U4、CentOS7U5、CentOS7U6、ESXi6.0、ESXi6.5、ESXi6.7、SLES11SP4、SLES12SP2、SLES12SP3、Ubuntu16.04、Ubuntu16.04.1、Ubuntu16.04.2、Win2016、Win2016 Standard Desktop、Win2016 Standard Core、Win2016 Datacenter Desktop、Win2016 Datacenter Core、Win2012\_R2、Win2012\_R2 Standard Desktop、Win2012\_R2 Standard Core、Win2012\_R2 Datacenter Desktop、Win2012\_R2 Datacenter Core、EulerOSV2SP3 |
| BootType | BIOS启动模式（选填） | * UEFIBoot * LegacyBoot * SecureBoot |
| CDKey | OS系统的安装秘钥（选填） | * Windows/Vmware：选填，取值为25位安装秘钥，每5位之间用‘-’连接，选值为[a-zA-Z0-9] * Linux：为空 |
| RootPwd | 管理员初始密码（必填） | 在myhosts文件中设置该参数，设置规则如下：   * Windows：最少6位 * SUSE：最少6位 * CentOS/Red Hat/EulerOS：最少6位，不含#$空格 * Ubuntu：最少8位，不含#$空格 * Vmware：最少7位，ESXi6.7版本至少需要字母、数字和特殊字符等三种不同类型字符，最多40位   说明  V119及以上版本的Smart Provisioning支持输入#$空格特殊字符。 |
| HostName | 主机名称（选填） | 取值规则：取值为[a-z,A-Z0-9-]，总长度不超过15个字符   * Linux：非必填，必须配置网络才可生效 * Windows：非必填 * Vmware：非必填，必须配置网络才可生效 |
| Language | 系统语言（必填） | 取值规则：字符串，请参考各个操作系统的安装指南   * Linux：必填 * Windows：必填 * Vmware：为空 |
| TimeZone | 系统时区（必填） | 取值规则：字符串，请参考各个操作系统的安装指南   * Linux：必填 * Windows：必填 * Vmware：保持为空 |
| Keyboard | 系统键盘类型（必填） | 取值规则：字符串，请参考各个操作系统的安装指南   * Linux：必填 * Windows：必填 * Vmware：为空 |
| CheckFirmware | 是否校验固件（必填） | * true * false |
| Partition | 分区信息（选填）  格式为：  Partition:  - Size: "64"  FileSystem: "NTFS"  Name: "C" | * Windows：Name的取值为C-Z的字符；FileSystem的取值为NTFS；Size的取值大于32，如果填max，则将整盘作为数据盘 * Linux：Name的取值不包含<>|:&和空格，如/、 /home、swap等；FileSystem的取值为ext4、ext3、ext2和xfs；Size的取值大于0，其中根分区需要大于10，swap分区需要大于1，如果填max，则分配剩余的空间 * Vmware：不支持 |
| Autopart | 是否支持自动分区（必填） | * Linux/Vmware：选值为true * Window：选值为false，不支持自动分区 |
| MediaType | 支持系统部署的介质类型（不选该字段表示部署在本地硬盘上）（选填） | * SANBoot：选择在SANBoot盘部署操作系统（仅支持VMware 6.5.1和VMware 6.7） * Disk：选择在Disk设备上部署操作系统 * USB：选择在USB设备上部署操作系统（仅支持VMware 6.5） |
| AutoPosition | 是否支持自动选择安装盘（必填） | true（当前仅支持自动选择安装盘） |
| Software | 需要安装的软件列表（必填）  格式为：  Software:  - FileName: "iBMA" | iBMA |
| NetCfg | 网络配置（选填） | []或配置以下参数：  说明  []表示不指定设备，可批量部署。   * Device：设备网络信息   + - Name：被部署服务器网卡网口名称     - MAC：设备MAC地址 * IPv4Addresses：网口的IPv4地址信息   + - Address：IPv4地址     - SubnetMash：子网掩码     - AddressOrigin：IPv4地址获取模式，可设置为“Static”或“DHCP”     - Gateway：IPv4网关地址 * IPv6Addresses：网口的IPv6地址信息   + - Address：IPv6地址     - PrefixLength：IPv6地址的前缀长度     - AddressOrigin：IPv6地址获取模式，可设置为“Static”或“DHCP”     - Gateway：IPv6网关地址   说明  Ubuntu和VMware系统不支持配置该选项。   * NameServers：DNS服务器的IP地址，取值可以为IPv4或IPv6的网络地址 |

OSType、Language、TimeZone和Keyboard的取值示例如下：

| OSType | Language | TimeZone | Keyboard |
| --- | --- | --- | --- |
| RHEL/CentOS/EulerOS/Ubuntu | en\_US.UTF-8 | America/New\_York | us |
| SLES | en\_US | America/New\_York | english-us |
| Windows | en-US | Eastern Standard Time | 0x00000409 |

执行部署

1. 进入“/home/ibmc\_ansible/examples”文件目录。

**cd /home/ibmc\_ansible/examples**

1. 执行配置命令。

**ansible-playbook deploy\_centos7u3\_by\_sp.yml**

如下返回信息表示执行成功。

[root@localhost examples]# ansible-playbook deploy\_centos7u3\_by\_sp.yml   
   
PLAY [ibmc deploy centos7u3 by sp] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [ibmc deploy centos7u3 by sp] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host1.domain.com]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host1.domain.com : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

1. 如何获取帮助
   1. 联系华为前准备

如果在日常维护或故障处理过程中遇到难以解决或者重大问题的时候，请寻求华为公司的技术支持。

为了更好的解决故障，建议在寻求华为技术支持前做好如下准备。

收集必要的故障信息

收集的信息包括：

* 客户的详细名称、地址
* 联系人姓名、电话号码
* 故障发生的具体时间
* 故障现象的详细描述
* 设备类型及软件版本
* 故障后已采取的措施和结果
* 问题的级别及希望解决的时间

做好必要的调试准备

在寻求华为技术支持时，华为技术支持工程师可能会协助您做一些操作，以进一步收集故障信息或者直接排除故障，所以在寻求技术支持前请收集必要的故障信息，准备好服务器刀片和扣卡的备件、螺丝刀、螺丝、串口线、网线等可能使用到的物品。

* 1. 如何从网站获取帮助

华为技术有限公司通过办事处、公司二级技术支持体系、电话技术指导、远程支持及现场技术支持等方式向用户提供及时有效的技术支持。

华为公司技术支持体系包括：

* 华为总部技术支持部
* 办事处技术支持中心
* 技术支持网站
* 客户服务中心

华为技术支持网站：<http://support.huawei.com/enterprise>

请您参照以下方法查阅最新的产品手册：

1. 登录<http://support.huawei.com/enterprise>。
2. 单击“登录”进入“用户登录”界面。
3. 输入您的帐号、密码和随机校验码后单击“登录”进入“技术支持首页”。
4. 在导航树上选择“技术支持 > 技术支持 > 产品和解决方案支持”，根据产品名称选择相应的产品手册。



您也可以在界面右上角通过搜索关键字的方式快速找到最新的产品手册。