|  |  |  |
| --- | --- | --- |
|  |  | |
|  | | |
|  | |  |
|  | | |
| Huawei Ansible Plug-in | |  |
| **用户指南** | |  |
|  | | |
| V2.0.6 | |  |
|  | | |
| 文档版本:01  发布日期:2021-06-30 | |  |
|  | | |
|  | 版权所有©版权所有权人 | |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | | |
|  | | | |
|  | 注意 |  |
|  | 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。 |

目 录

[前言 1](#_Toc256000000)

[1 简介 1-1](#_Toc256000001)

[2 安装和卸载华为Ansible插件 2-1](#_Toc256000002)

[2.1 安装华为Ansible插件 2-1](#_Toc256000003)

[2.2 卸载华为Ansible插件 2-2](#_Toc256000004)

[3 配置华为Ansible插件 3-1](#_Toc256000005)

[3.1 配置/etc/ansible/hosts文件 3-1](#_Toc256000006)

[3.2 配置/group\_vars/myhosts文件 3-2](#_Toc256000007)

[3.3 配置SSL证书认证和TLS 1.2通讯方式 3-5](#_Toc256000008)

[4 使用华为Ansible插件 4-1](#_Toc256000009)

[4.1 查询服务器基本信息 4-5](#_Toc256000010)

[4.2 管理设备启动项 4-6](#_Toc256000011)

[4.2.1 查询启动项配置信息 4-6](#_Toc256000012)

[4.2.2 配置启动设备 4-7](#_Toc256000013)

[4.3 管理电源 4-8](#_Toc256000014)

[4.3.1 查询电源状态 4-8](#_Toc256000015)

[4.3.2 配置电源状态 4-9](#_Toc256000016)

[4.4 管理iBMC用户 4-10](#_Toc256000017)

[4.4.1 查询iBMC用户（生成json文件） 4-10](#_Toc256000018)

[4.4.2 创建iBMC用户 4-10](#_Toc256000019)

[4.4.3 修改iBMC用户 4-12](#_Toc256000020)

[4.4.4 删除iBMC用户 4-14](#_Toc256000021)

[4.5 iBMC网络配置 4-15](#_Toc256000022)

[4.5.1 查询iBMC网络信息（生成json文件） 4-15](#_Toc256000023)

[4.5.2 配置iBMC网络信息 4-16](#_Toc256000024)

[4.6 管理NTP服务 4-17](#_Toc256000025)

[4.6.1 查询NTP服务信息 4-18](#_Toc256000026)

[4.6.2 配置NTP 4-18](#_Toc256000027)

[4.7 管理SNMP Trap服务 4-20](#_Toc256000028)

[4.7.1 查询SNMP服务信息（生成json文件） 4-20](#_Toc256000029)

[4.7.2 配置SNMP Trap 4-20](#_Toc256000030)

[4.8 导入导出Profile文件 4-23](#_Toc256000031)

[4.8.1 导入Profile文件 4-23](#_Toc256000032)

[4.8.2 导出Profile文件 4-25](#_Toc256000033)

[4.9 固件升级 4-27](#_Toc256000034)

[4.9.1 查询固件版本信息（生成json文件） 4-27](#_Toc256000035)

[4.9.2 升级固件 4-28](#_Toc256000036)

[4.9.2.1 带外固件升级 4-28](#_Toc256000037)

[4.9.2.2 带内固件升级 4-30](#_Toc256000038)

[4.10 RAID配置 4-32](#_Toc256000039)

[4.10.1 查询RAID配置（生成json文件） 4-32](#_Toc256000040)

[4.10.2 删除RAID组 4-33](#_Toc256000041)

[4.10.3 创建RAID组 4-34](#_Toc256000042)

[4.10.4 修改RAID配置 4-36](#_Toc256000043)

[4.11 OS部署 4-38](#_Toc256000044)

[4.11.1 ServiceCD2.0方式 4-38](#_Toc256000045)

[4.11.2 Smart Provisioning方式 4-42](#_Toc256000046)

[4.12 BIOS管理 4-48](#_Toc256000047)

[4.12.1 查询BIOS信息（生成json文件） 4-48](#_Toc256000048)

[4.12.2 设置BIOS信息 4-50](#_Toc256000049)

[4.12.3 恢复BIOS默认配置 4-51](#_Toc256000050)

[4.13 日志管理 4-52](#_Toc256000051)

[4.13.1 一键收集iBMC日志 4-52](#_Toc256000052)

[4.13.2 收集SEL日志 4-54](#_Toc256000053)

[4.13.3 清空SEL日志 4-55](#_Toc256000054)

[4.14 通用接口 4-56](#_Toc256000055)

[4.15 文件本地传输 4-57](#_Toc256000056)

[4.15.1 上传本地文件 4-58](#_Toc256000057)

[4.15.2 下载文件至本地 4-59](#_Toc256000058)

[A FAQ 1](#_Toc256000059)

[B 获取技术支持 1](#_Toc256000060)

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 | 2021-06-30 | 第一次正式发布。 |

# 简介

功能介绍

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）
* 管理BIOS（包括查询和设置BIOS信息、恢复BIOS默认配置）
* 管理日志（包括一键收集iBMC日志、收集SEL日志、清空SEL日志）
* 提供通用公共接口
* 上传本地文件
* 下载文件至本地

支持的服务器

| 架构 | 类型 | 型号 |
| --- | --- | --- |
| x86 | 机架服务器 | RH2288H V3 |
| 2488 V5 |
| 2288H V5 |
| 1288H V6 |
| 2288H V6 |
| 5288 V6 |
| 刀片服务器 | CH121 V3 |
| CH242 V3 DDR4 |
| CH121 V5 |
| CH242 V5 |
| CH121L V5 |
| MM921 |
| CX621 |
| CX320 |
| 高密服务器 | XH622 V3 |
| XH321 V5 |
| 异构服务器 | G560 V5 |
| Arm | 机架服务器 | TaiShan 200服务器（型号2280） |
| TaiShan 200服务器（型号2280, VE） |
| TaiShan 200K服务器（型号2280K） |
| TaiShan 200服务器（型号5280） |
| TaiShan 200服务器（型号5280, VE） |
| TaiShan 200K服务器（型号5280K） |
| TaiShan 200服务器（型号5290） |
| TaiShan 200服务器（型号2480） |
| TaiShan 200K服务器（型号2480K） |
| TaiShan 200服务器（型号2180） |
| TaiShan 200服务器（型号5180） |
| TaiShan 200 Pro服务器（型号2480） |

版本配套关系

| 版本名称 | 配套版本 |
| --- | --- |
| 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 |

# 安装和卸载华为Ansible插件

软件要求

* Ansible：2.5.0及以上（推荐使用2.10以上版本）
* Python：2.7及以上、3.7及以上



需要在Python环境上安装requests-toolbelt依赖包（0.9.1及以上版本），以支撑部分功能的使用。

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

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

## 安装华为Ansible插件

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

以root用户登录Ansible服务器。

将安装包上传到Ansible服务器的root用户目录。

进入华为Ansible插件安装包的所在目录。

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，该文件夹下保存了华为Ansible插件的SSL配置文件，以及在/home/ibmc\_ansible/examples文件夹下保存了执行华为Ansible插件时需要配置的yml样例文件。

----结束

## 卸载华为Ansible插件

以root用户登录Ansible服务器。

进入华为Ansible插件安装包的所在目录。

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

执行卸载命令**。**

**python uninstall.py**

在提示是否保存yml文件时，根据实际需要输入“n”（不保存）或者“y”（保存），此处以不保存为例。

[root@localhost Huawei\_iBMC\_Ansible\_Module]# python uninstall.py   
start uninstalling Huawei\_ibmc\_ansible module   
do you want to keep the yml files?(y/n)   
**n**

在提示是否保存日志文件和插件生成的文件时，根据实际需要输入“n”（不保存）或者“y”（保存），此处以不保存为例。

[root@localhost Huawei\_iBMC\_Ansible\_Module]# python uninstall.py   
start uninstalling Huawei\_ibmc\_ansible module   
do you want to keep the yml files?(y/n)   
n   
do you want to keep the log files and plug-in generation file?(y/n)   
**n**   
rm ibmc\_ansible log successfully!

插件卸载成功后会提示卸载成功。

[root@localhost Huawei\_iBMC\_Ansible\_Module]# python uninstall.py   
start uninstalling Huawei\_ibmc\_ansible module   
do you want to keep the yml files?(y/n)   
n   
do you want to keep the log files and plug-in generation file?(y/n)   
n   
rm ibmc\_ansible log successfully!   
**uninstalling Huawei\_ibmc\_ansible successfully!**

----结束

# 配置华为Ansible插件



本章节中的文件使用加密方式创建。加密文件、查看或设置已加密文件、以及文件加密后，如何执行配置命令的具体步骤可参见A.1 如何加密文件以及如何查看、编辑和执行已加密的文件。

[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

执行以下命令，创建/etc/ansible。

**mkdir /etc/ansible**

执行以下命令，进入/etc/ansible目录。

**cd /etc/ansible**

使用以下命令创建“hosts”文件。

**vi hosts**

将myhost信息写入“hosts”文件。

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

----结束



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



可使用root用户或者非root用户进行配置，因涉及到密码等敏感数据，推荐使用非root用户进行配置。

设置“/home/*用户*/ibmc\_ansible/examples/group\_vars”（非root用户）或者“/home/ibmc\_ansible/examples/group\_vars”（root用户）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。



“/home/*用户*/ibmc\_ansible/examples/group\_vars”：“用户”为实际的非root用户名称，本文档以用户“plugin”为例。

操作步骤（非root用户）

执行以下命令切换用户权限。

**su plugin**

执行以下命令，拷贝/home目录下的ibmc\_ansible文件夹至/home/plugin目录下。

**cp -r /home/ibmc\_ansible/ /home/plugin/ibmc\_ansible/**

执行以下命令，创建/home/plugin/ibmc\_ansible/examples/group\_vars目录。

**mkdir /home/plugin/ibmc\_ansible/examples/group\_vars**

执行以下命令，进入/home/plugin/ibmc\_ansible/examples/group\_vars目录。

**cd /home/plugin/ibmc\_ansible/examples/group\_vars**

使用加密命令创建“myhosts”文件。

**ansible-vault create myhosts**



* 因涉及到密码等敏感数据，推荐对“myhosts”文件进行加密创建。加密创建后，执行命令时需要使用**--ask-vault-pass**进行解密，具体请参见A.1 如何加密文件以及如何查看、编辑和执行已加密的文件。
* 非加密方式创建命令为**create myhosts**，该方式可能会导致密码等敏感数据的泄露，请谨慎使用。

在“myhosts”文件中写入以下内容。

# 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"**

----结束

操作步骤（root用户）

执行以下命令，创建/home/ibmc\_ansible/examples/group\_vars目录。

**mkdir /home/ibmc\_ansible/examples/group\_vars**

执行以下命令，进入/home/ibmc\_ansible/examples/group\_vars目录。

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

使用加密命令创建“myhosts”文件。

**ansible-vault create myhosts**



* 因涉及到密码等敏感数据，推荐对“myhosts”文件进行加密创建。加密创建后，执行命令时需要使用**--ask-vault-pass**进行解密，具体请参见A.1 如何加密文件以及如何查看、编辑和执行已加密的文件。
* 非加密方式创建命令为**create myhosts**，该方式可能会导致密码等敏感数据的泄露，请谨慎使用。

在“myhosts”文件中写入以下内容。

# 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"**

----结束

## 配置SSL证书认证和TLS 1.2通讯方式

参数配置

修改“/home/plugin/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”。



关闭force\_tls1\_2（设置为“False”）会存在安全风险，请谨慎操作。

* 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

* 使用Python安装certifi证书库时，默认使用此证书库，如：

Python2环境下：/usr/lib/python2.7/site-packages/certifi-2019.11.28-py2.7.egg/certifi/cacert.pem

Python3环境下：/usr/local/python3/lib/python3.7/site-packages/certifi-2020.6.20-py3.7.egg/certifi/cacert.pem



关闭SSL证书认证（“verify”设置为“False”）会存在安全风险，请谨慎操作。

* ciphers：Ansible插件作为客户端与服务器建立会话时使用的加密套件。



建议使用安全的加密套件，使用不安全的加密套件存在安全风险，请谨慎使用。

[plugin@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: False**   
 **verify: False**   
 **certify:**   
 **ciphers: "ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:DHE-DSS-AES128-GCM-SHA256:DHE-DSS-AES256-GCM-SHA384:ECDHE-RSA-CHACHA20-POLY1305"**

执行命令

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

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

1. 执行配置命令。

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



该命令只能使用root用户执行，使用非root用户时无权限执行。

如下返回信息表示执行成功。

[plugin@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



* 本章中的文件均以未加密文件为例，建议为涉及密码等敏感数据的文件进行加密。加密文件、查看或设置已加密文件、以及文件加密后，如何执行配置命令的具体步骤请参见A.1 如何加密文件以及如何查看、编辑和执行已加密的文件。
* 本文档使用非root用户“plugin”为例进行执行操作说明。
* 当Ansible插件配套的iBMC版本为做了HTTPS证书校验的iBMC版本时，Ansible插件无法使用HTTPS方式的远程传输方式，建议使用其他协议。

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，此处以Python3环境为例。

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

[root@localhost ~]# ansible-doc -l |grep ibmc   
/usr/local/python3/lib/python3.7/site-packages/cryptography-3.1.1-py3.7-linux-x86\_64.egg/cryptography/hazmat/bindings/openssl/binding.py:177: CryptographyDeprecationWarning: OpenSSL version 1.0.2 is no longer supported by the OpenSSL project, please upgrade. The next version of cryptography will drop support for it.   
 utils.CryptographyDeprecationWarning,   
ibmc\_ansible\_show\_version Show Huawei iBMC ansible modules version   
ibmc\_clear\_sel\_logs Clear SEL logs   
ibmc\_collect\_logs Collect iBMC logs   
ibmc\_collect\_sel\_logs Collect iBMC SEL logs   
ibmc\_common\_api Common api   
ibmc\_create\_account Create an ibmc user   
ibmc\_create\_raid Create volume   
ibmc\_delete\_account Delete an ibmc user   
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\_download\_file Download files   
ibmc\_get\_account Get ibmc user info   
ibmc\_get\_basic\_info Get server information   
ibmc\_get\_bios Get bios 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 an ibmc user   
ibmc\_modify\_raid Modify volume   
ibmc\_outband\_fw\_update update outband firmware   
ibmc\_profile\_export export the server profile   
ibmc\_profile\_import import the server profile   
ibmc\_reset\_bios Reset BIOS resource attributes   
ibmc\_set\_bios Set bios info   
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\_redfish\_request\_cfg set request config   
ibmc\_set\_snmp\_trap Set snmp trap info   
ibmc\_upload\_file upload file

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   
/usr/local/python3/lib/python3.7/site-packages/cryptography-3.1.1-py3.7-linux-x86\_64.egg/cryptography/hazmat/bindings/openssl/binding.py:177: CryptographyDeprecationWarning: OpenSSL version 1.0.2 is no longer supported by the OpenSSL project, please upgrade. The next version of cryptography will drop support for it.   
 utils.CryptographyDeprecationWarning,   
> IBMC\_GET\_ACCOUNT (/usr/local/python3/lib/python3.7/site-packages/ansible-2.9.9-py3.7.egg/ansible/modules/i   
   
 Get ibmc user 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:   
   
:...skipping...   
> IBMC\_GET\_ACCOUNT (/usr/local/python3/lib/python3.7/site-packages/ansible-2.9.9-py3.7.egg/ansible/modules/i   
   
 Get ibmc user 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/plugin/ibmc\_ansible/examples”文件目录。

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

1. 执行以下命令查看版本信息。

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

[plugin@localhost examples]# ansible-playbook -v show\_ibmc\_ansible\_version.yml   
/usr/local/python3/lib/python3.7/site-packages/cryptography-3.1.1-py3.7-linux-x86\_64.egg/cryptography/hazmat/bindings/openssl/binding.py:177: CryptographyDeprecationWarning: OpenSSL version 1.0.2 is no longer supported by the OpenSSL project, please upgrade. The next version of cryptography will drop support for it.   
 utils.CryptographyDeprecationWarning,   
No config file found; using defaults   
   
PLAY [show Huawei iBMC ansible modules version] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [show Huawei iBMC ansible modules version] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [127.0.0.1] => {"changed": false, "msg": "**Huawei iBMC ansible modules version is 2.0.6**"}   
   
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](#_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)

[4.12 BIOS管理](#_ZH-CN_TOPIC_0000001167612627)

[4.13 日志管理](#_ZH-CN_TOPIC_0000001167612631)

[4.14 通用接口](#_ZH-CN_TOPIC_0000001167772605)

[4.15 文件本地传输](#_ZH-CN_TOPIC_0000001120895452)

## 查询服务器基本信息

功能介绍

* 支持查询服务器BMC版本、BIOS版本、CPLD版本、Smart Provisioning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、健康状态信息。
* 默认生成json文件，若需生成CSV文件，在执行查询命令之前，需先配置get\_basic\_info.yml文件中的“csv\_format”为“True”。

操作步骤

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

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

1. 执行查询服务器基本信息命令。

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

如下返回信息表示命令执行成功。

[plugin@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，建议导出文件后再查看。

## 管理设备启动项



管理模块MM921、交换模块CX320/CX621不支持此功能。

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/plugin/ibmc\_ansible/examples”文件目录。

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

1. 执行查询启动项配置信息命令。

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



若需直接在返回信息中查看查询结果，可执行**ansible-playbook -vv get\_boot\_device.yml**命令进行查询。

如下返回信息表示命令执行成功。

[plugin@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

查询结果可在“/var/log/ansible/ibmc/report”目录下的“ansibleibmc.report”文件中进行查看，如[3](#d0e1398)所示。

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/plugin/ibmc\_ansible/examples/set\_boot\_device.yml”文件。

[plugin@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"**

执行命令

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

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

1. 执行配置启动设备命令。

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

如下返回信息表示命令执行成功。

[plugin@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



管理模块MM921、交换模块CX320/CX621不支持此功能。

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/plugin/ibmc\_ansible/examples”文件目录。

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

1. 执行查询电源状态命令。

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

如下返回信息表示命令执行成功。

[plugin@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/plugin/ibmc\_ansible/examples/set\_power.yml”文件。

[plugin@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"**

执行命令

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

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

1. 执行配置电源状态命令。

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

如下返回信息表示命令执行成功。

[plugin@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/plugin/ibmc\_ansible/examples”文件目录。

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

1. 执行查询iBMC用户命令。

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

如下返回信息表示命令执行成功。

[plugin@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/plugin/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/plugin/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/plugin/ibmc\_ansible/examples/create\_account.yml”文件。

[plugin@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"**

执行命令

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

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

1. 执行创建iBMC用户命令。

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

如下返回信息表示命令执行成功。

[plugin@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/plugin/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。

[plugin@localhost examples]# vi /home/plugin/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/plugin/ibmc\_ansible/examples/modify\_account.yml”文件。

[plugin@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**

执行命令

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

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

1. 执行修改iBMC用户命令。

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

如下返回信息表示命令执行成功。

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

[plugin@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"**

执行命令

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

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

1. 执行删除iBMC用户命令。

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

如下返回信息表示命令执行成功。

[plugin@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



管理模块MM921、交换模块CX320/CX621不支持此功能。

功能介绍

* 支持查询iBMC网络信息。
* 支持单独修改IP使能模式或IP地址信息。



* 不能同时修改IP使能模式和IP地址信息，否则会出现修改不成功的现象。
* IPv4使能模式和IPv6使能模式之间不能相互跳转，否则会造成服务器无法连接的问题。

### 查询iBMC网络信息（生成json文件）

操作步骤

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

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

1. 执行查询iBMC网络信息命令。

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

如下返回信息表示命令执行成功。

[plugin@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



Ansible插件目前仅支持IPv4连接到服务器，建议不要关闭IPv4使能模式。

参数配置

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

[plugin@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"**

执行命令

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

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

1. 执行配置iBMC网络信息命令。

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

如下返回信息表示命令执行成功。

[plugin@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



管理模块MM921、交换模块CX320/CX621不支持此功能。

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/plugin/ibmc\_ansible/examples”文件目录。

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

1. 执行查询NTP服务信息命令。

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

如下返回信息表示命令执行成功。

[plugin@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/plugin/ibmc\_ansible/examples/set\_ntp.yml”文件。

[plugin@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**

执行命令

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

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

1. 执行配置NTP命令。

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

如下返回信息表示命令执行成功。

[plugin@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



管理模块MM921、交换模块CX320/CX621不支持此功能。

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/plugin/ibmc\_ansible/examples”文件目录。

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

1. 执行查询SNMP服务信息命令。

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

如下返回信息表示命令执行成功。

[plugin@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/plugin/ibmc\_ansible/examples/set\_snmp\_trap.yml”文件。

[plugin@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/plugin/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/plugin/ibmc\_ansible/examples/group\_vars/myhosts”文件中配置“community”参数。

[plugin@localhost examples]# vi /home/plugin/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"

执行命令

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

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

1. 执行配置SNMP Trap命令。

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

如下返回信息表示命令执行成功。

[plugin@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



管理模块MM921、交换模块CX320/CX621不支持此功能。

功能介绍

支持BIOS和iBMC的Profile配置文件的导入和导出。

* 支持使用本地或远程的方式将Profile文件导入到服务器。
* 支持将服务器的Profile文件导出到本地环境或者远程路径上。

### 导入Profile文件

参数配置

修改“/home/plugin/ibmc\_ansible/examples/profile\_import.yml”文件。

**本地导入文件时，配置的参数如下：**

* file\_name：需导入文件名称。
* local\_import：需导入的Ansible环境下的本地文件路径。

**远程导入文件时，配置的参数如下：**

* file\_name：需导入文件名称。
* remote\_file：需导入的远程目录下的文件路径。
* iBMC临时“/tmp”目录。
* 远程文件服务器目录下的文件夹，文件路径输入形式为“文件传输协议://远程文件服务器IP地址/文件夹”。

支持的文件传输协议包括sftp、https、nfs、cifs、scp。

* file\_server\_user：远程文件服务器用户名。
* file\_server\_pswd：远程文件服务器密码。

**本地导入时，配置的参数如下：**

[plugin@localhost examples]# vi profile\_import.yml   
   
- hosts: myhosts   
 connection: local   
 name: import profile   
 gather\_facts: False   
   
# file\_name: the file name you want to import   
# local\_import: local file path of the Ansible environment to be imported.   
# remote\_import: remote path for saving imported files. The file path can be /tmp on the BMC; or a folder on a remote file server, the format is protocol://ip/folder   
# protocols: Available values: sftp,https,nfs,cifs,scp   
# file\_server\_user: remote file server user name   
# file\_server\_pswd: remote file server password   
   
 tasks:   
 - name: import profile   
 ibmc\_profile\_import:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **file\_name: "192.168.1.1\_20210318045050\_profile.xml"**   
 **local\_import: "/home"**

**远程导入时，配置的参数如下：**

[plugin@localhost examples]# vi profile\_import.yml   
   
- hosts: myhosts   
 connection: local   
 name: import profile   
 gather\_facts: False   
   
# file\_name: the file name you want to import   
# local\_import: local file path of the Ansible environment to be imported.   
# remote\_import: remote path for saving imported files. The file path can be /tmp on the BMC; or a folder on a remote file server, the format is protocol://ip/folder   
# protocols: Available values: sftp,https,nfs,cifs,scp   
# file\_server\_user: remote file server user name   
# file\_server\_pswd: remote file server password   
   
 tasks:   
 - name: import profile   
 ibmc\_profile\_import:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **file\_name: "192.168.1.1\_20210318045050\_profile.xml"**   
 **remote\_import: "sftp://192.168.1.1/data/"**   
 **file\_server\_user: "{{sftp\_user}}"**   
 **file\_server\_pswd: "{{sftp\_pswd}}"**

执行命令

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

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

1. 执行导入Profile文件命令。

**ansible-playbook profile\_import.yml**

如下返回信息表示命令执行成功。

[plugin@localhost examples]# ansible-playbook profile\_import.yml   
/usr/local/python3/lib/python3.7/site-packages/cryptography-3.1.1-py3.7-linux-x86\_64.egg/cryptography/hazmat/bindings/openssl/binding.py:177: CryptographyDeprecationWarning: OpenSSL version 1.0.2 is no longer supported by the OpenSSL project, please upgrade. The next version of cryptography will drop support for it.   
 utils.CryptographyDeprecationWarning,   
   
PLAY [import profile] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [import profile] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

### 导出Profile文件

参数配置

修改“/home/plugin/ibmc\_ansible/examples/profile\_export.yml”文件。

**导出文件至Ansible环境上时，配置的参数如下：**

* file\_name：文件导出后的名称，可选参数。如不配置该参数，默认导出后文件名称为“*IP地址*\_profile.xml”。
* local\_export：用于保存导出文件的Ansible环境下的本地路径。

**导出文件至远程路径时，配置的参数如下：**

* file\_name：文件导出后的名称，可选参数。如不配置该参数，默认导出后文件名称为“*IP地址*\_profile.xml”。
* remote\_export：用于保存导出文件的远程路径。
* iBMC临时“/tmp”目录。
* 远程文件服务器目录下的文件夹，文件路径输入形式为“文件传输协议://远程文件服务器IP地址/文件夹”。

支持的文件传输协议包括sftp、https、nfs、cifs、scp。

* file\_server\_user：远程文件服务器用户名。
* file\_server\_pswd：远程文件服务器密码。

**导出文件至Ansible环境上时，配置的参数如下：**

[plugin@localhost examples]# vi profile\_export.yml   
   
- hosts: myhosts   
 connection: local   
 name: import profile   
 gather\_facts: False   
   
# file\_name: the file name you want to import   
# local\_import: local file path of the Ansible environment to be imported.   
# remote\_import: remote path for saving imported files. The file path can be /tmp on the BMC; or a folder on a remote file server, the format is protocol://ip/folder   
# protocols: Available values: sftp,https,nfs,cifs,scp   
# file\_server\_user: remote file server user name   
# file\_server\_pswd: remote file server password   
   
 tasks:   
 - name: import profile   
 ibmc\_profile\_import:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **file\_name: "192.168.1.1\_20210318045050\_profile.xml"**   
 **local\_export: "/home"**

**导出文件至远程路径时，配置的参数如下：**

[plugin@localhost examples]# vi profile\_export.yml   
   
- hosts: myhosts   
 connection: local   
 name: import profile   
 gather\_facts: False   
   
# file\_name: the file name you want to import   
# local\_import: local file path of the Ansible environment to be imported.   
# remote\_import: remote path for saving imported files. The file path can be /tmp on the BMC; or a folder on a remote file server, the format is protocol://ip/folder   
# protocols: Available values: sftp,https,nfs,cifs,scp   
# file\_server\_user: remote file server user name   
# file\_server\_pswd: remote file server password   
   
 tasks:   
 - name: import profile   
 ibmc\_profile\_import:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **file\_name: "192.168.1.1\_20210318045050\_profile.xml"**   
 **remote\_export: "sftp://192.168.1.1/data/"**   
 **file\_server\_user: "{{sftp\_user}}"**   
 **file\_server\_pswd: "{{sftp\_pswd}}"**

执行命令

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

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

1. 执行导出Profile文件命令。

**ansible-playbook profile\_import.yml**

如下返回信息表示命令执行成功。

[plugin@localhost examples]# ansible-playbook profile\_export.yml   
/usr/local/python3/lib/python3.7/site-packages/cryptography-3.1.1-py3.7-linux-x86\_64.egg/cryptography/hazmat/bindings/openssl/binding.py:177: CryptographyDeprecationWarning: OpenSSL version 1.0.2 is no longer supported by the OpenSSL project, please upgrade. The next version of cryptography will drop support for it.   
 utils.CryptographyDeprecationWarning,   
   
PLAY [export profile] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [export profile] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0 : 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。
* 管理模块MM921、交换模块CX320/CX621不支持此功能。

操作步骤

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

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

1. 执行查询固件版本信息命令。

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

如下返回信息表示命令执行成功。

[plugin@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/plugin/ibmc\_ansible/examples/update\_outband\_fw.yml”文件。

**使用本地的升级文件时，配置的参数如下：**

local\_file：本地升级文件的路径，输入形式为“目录/文件名”，如/home/cpldimage.hpm。

**使用远程目录的升级文件时，配置的参数如下：**

* remote\_file：远程目录升级文件的路径。
* 使用iBMC临时目录时，必须为“/tmp”目录，而且要指定一个文件名，如/tmp/cpldimage.hpm。
* 使用远程文件服务器目录时，文件路径输入形式为“文件传输协议://远程文件服务器IP地址/目录/文件名”，如sftp://192.168.1.1/data/cpldimage.hpm。

支持的文件传输协议包括sftp、https、nfs、cifs、scp。

* file\_server\_user：远程文件服务器用户名。
* file\_server\_pswd：远程文件程服务器密码。



* 交换模块CX320和CX621仅支持SFTP协议，管理模块MM921仅支持SFTP和NFS协议。
* 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。

**使用本地的升级文件时，配置如下：**

[plugin@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 }}"   
 **local\_file: "/home/cpldimage.hpm"**

**使用远程目录的升级文件时，配置如下：**

[plugin@localhost examples]# vi update\_outband\_fw.yml   
---   
- hosts: myhosts   
 connection: local   
 name: update outband fw   
 gather\_facts: False   
   
# local\_file: the local firmware file needs to be upgraded, the format is directory/filename, for example: /home/cpldimage.hpm.   
# remote\_file: remote firmware file. The firmware file can be /tmp on the BMC, the format is directory/filename, for example: /tmp/cpldimage.hpm;   
 # or a firmware file on a remote file server, the format is protocol://ip/folder/filename, for example: sftp://172.26.200.11/data/cpldimage.hpm.   
# protocols: Available values: sftp,https,nfs,cifs,scp   
# file\_server\_user: remote file server user name   
# file\_server\_pswd: remote file server password   
   
 tasks:   
 - name: update outband fw   
 ibmc\_outband\_fw\_update:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **remote\_file: "sftp://192.168.1.1/data/cpldimage.hpm"**   
 **file\_server\_user: "{{sftp\_user}}"**   
 **file\_server\_pswd: "{{sftp\_pswd}}"**

执行命令

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

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

1. 执行升级命令。

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

如下返回信息表示命令执行成功。

[plugin@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



管理模块MM921、交换模块CX320/CX621不支持此功能。

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/plugin/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。

[plugin@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/plugin/ibmc\_ansible/examples/group\_vars”目录下的“myhosts”文件中修改。

[plugin@localhost examples]# vi /home/plugin/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"

执行命令

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

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

1. 执行升级命令。

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

如下返回信息表示命令执行成功。

[plugin@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



管理模块MM921、交换模块CX320/CX621不支持此功能。

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/plugin/ibmc\_ansible/examples”文件目录。

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

1. 执行查询RAID配置命令。

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

如下返回信息表示命令执行成功。

[plugin@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/plugin/ibmc\_ansible/examples/delete\_raid.yml”文件。

[plugin@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"**

执行命令

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

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

1. 执行删除RAID组命令。

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

如下返回信息表示命令执行成功。

[plugin@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/plugin/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)。

[plugin@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 number. It is a mandatory parameter. Format: "1,2,.,N"   
# 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"**

执行命令

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

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

1. 执行创建RAID组命令。

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

如下返回信息表示命令执行成功。

[plugin@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/plugin/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)。

[plugin@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"**

执行命令

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

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

1. 执行修改RAID配置命令。

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

如下返回信息表示命令执行成功。

[plugin@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



* 管理模块MM921、交换模块CX320/CX621不支持OS部署功能。
* 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/plugin/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。

[plugin@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 |

执行命令

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

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

1. 执行部署命令。

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

如下返回信息表示命令执行成功。

[plugin@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/plugin/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。

[plugin@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: "\*\*:\*\*:\*\*:\*\*:\*\*:\*\*"**   
 **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 |

执行命令

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

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

1. 执行部署命令。

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

如下返回信息表示命令执行成功。

[plugin@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

## BIOS管理

功能介绍

支持查询和设置BIOS信息，以及恢复BIOS默认配置。

### 查询BIOS信息（生成json文件）

参数配置（可选）

修改“/home/plugin/ibmc\_ansible/examples/get\_bios.yml”文件的“bios\_attribute”参数。

bios\_attribute：查询指定的BIOS参数项信息，可选参数。若不配置该参数，则查询所有BIOS参数项信息。

查询后生成“*IP地址*\_BIOSInfo.json”文件，保存在/home/*用户*/ansible\_ibmc/report/bios/文件夹中。



* *IP地址*\_BIOSInfo.json：“IP地址”为配置的服务器IP地址。
* /home/*用户*/ansible\_ibmc/report/bios/：“用户”为实际使用的用户名。

[plugin@localhost examples]# vi get\_bios.yml   
   
---   
- hosts: myhosts   
 connection: local   
 name: get bios   
 gather\_facts: False   
   
# bios\_attribute: User-specified BIOS attributes to be queried   
   
 tasks:   
 - name: get bios   
 ibmc\_get\_bios:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **bios\_attribute:**   
 **- QuickBoot**   
 **- QuietBoot**

执行命令

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

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

1. 执行查询BIOS信息命令。

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

如下返回信息表示命令执行成功。

[plugin@localhost examples]# ansible-playbook get\_bios.yml   
/usr/local/python3/lib/python3.7/site-packages/cryptography-3.1.1-py3.7-linux-x86\_64.egg/cryptography/hazmat/bindings/openssl/binding.py:177: CryptographyDeprecationWarning: OpenSSL version 1.0.2 is no longer supported by the OpenSSL project, please upgrade. The next version of cryptography will drop support for it.   
 utils.CryptographyDeprecationWarning,   
   
PLAY [get bios] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [get bios] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

### 设置BIOS信息

参数配置

修改“/home/plugin/ibmc\_ansible/examples/set\_bios.yml”文件中的“Immediately”和“bios\_attribute”参数。

* Immediately：设置生效时间，取值为“True”和“False”，默认为“False”。“True”表示立即重启使设置生效，服务器会自动重启；“False”表示不会立即重启服务器，需要用户自行重启使设置生效。



服务器自动重启会影响业务，请谨慎操作。

* bios\_attribute：需要设置的BIOS参数项，必选项。



下面以设置“QuickBoot”和“QuietBoot”为例。关于可设置的BIOS参数项可参考《[华为服务器 Purley平台 BIOS 参数参考](https://support.huawei.com/enterprise/zh/doc/EDOC1000163371)》或者《[BIOS 参数参考 (鲲鹏920处理器)](https://support.huawei.com/enterprise/zh/doc/EDOC1100088653)》。

[plugin@localhost examples]# vi set\_bios.yml   
   
---   
- hosts: myhosts   
 connection: local   
 name: set ibmc bios   
 gather\_facts: False   
   
# Immediately: Whether to restart the system immediately for the configuration to take effect: True, False   
# bios\_attribute: BIOS attributes set by the user   
   
 tasks:   
 - name: set ibmc bios   
 ibmc\_set\_bios :   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **Immediately: False**   
 **bios\_attribute:**   
 **QuickBoot: Disabled**   
 **QuietBoot: Enabled**

执行命令

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

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

1. 执行设置BIOS信息命令。

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

如下返回信息表示命令执行成功。

[plugin@localhost examples]# ansible-playbook set\_bios.yml   
/usr/local/python3/lib/python3.7/site-packages/cryptography-3.1.1-py3.7-linux-x86\_64.egg/cryptography/hazmat/bindings/openssl/binding.py:177: CryptographyDeprecationWarning: OpenSSL version 1.0.2 is no longer supported by the OpenSSL project, please upgrade. The next version of cryptography will drop support for it.   
 utils.CryptographyDeprecationWarning,   
   
PLAY [set ibmc bios] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [set ibmc bios] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

### 恢复BIOS默认配置

参数配置

修改“/home/plugin/ibmc\_ansible/examples/reset\_bios.yml”文件的“Immediately”参数。

Immediately：设置生效时间，取值为“True”和“False”，默认为“False”。

* True：表示立即重启生效，服务器会自动重启。
* False：表示不会立即重启服务器，需要用户自行重启使设置生效。



服务器自动重启会影响业务，请谨慎操作。

[plugin@localhost examples]# vi reset\_bios.yml   
   
---   
- hosts: myhosts   
 connection: local   
 name: reset ibmc bios   
 gather\_facts: False   
   
# Immediately: Whether to restart the system immediately for the configuration to take effect: True, False   
   
 tasks:   
 - name: reset ibmc bios   
 ibmc\_reset\_bios :   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **Immediately: False**

执行命令

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

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

1. 执行重置BIOS信息命令。

**ansible-playbook reset\_bios.yml**

如下返回信息表示命令执行成功。

[plugin@localhost examples]# ansible-playbook reset\_bios.yml   
/usr/local/python3/lib/python3.7/site-packages/cryptography-3.1.1-py3.7-linux-x86\_64.egg/cryptography/hazmat/bindings/openssl/binding.py:177: CryptographyDeprecationWarning: OpenSSL version 1.0.2 is no longer supported by the OpenSSL project, please upgrade. The next version of cryptography will drop support for it.   
 utils.CryptographyDeprecationWarning,   
   
PLAY [reset ibmc bios] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [reset ibmc bios] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

## 日志管理

功能介绍

支持一键收集iBMC日志、收集SEL日志、清空SEL日志。

### 一键收集iBMC日志

参数配置

修改“/home/plugin/ibmc\_ansible/examples/collect\_ibmc\_logs.yml”文件。

* save\_mode：表示在远程文件服务器存储或者本地存储。远程文件服务器存储取值为“sftp”、“https”、“nfs”、“cifs”或“scp”；本地存储取值为“local”。
* file\_server\_ip：远程文件服务器IP地址，当“save\_mode”设置为“sftp”、“https”、“nfs”、“cifs”或“scp”时，需要设置。
* file\_server\_user：远程文件服务器用户名，当“save\_mode”设置为“sftp”、“https”、“cifs”或“scp”时，需要设置。
* file\_server\_pswd：远程文件服务器密码，当“save\_mode”设置为“sftp”、“https”、“cifs”或“scp”时，需要设置。
* file\_name：日志文件保存的路径与文件名。只设置文件名时，日志文件默认保存在/home/*用户*/ansible\_ibmc/report/collect\_IBMC\_log/路径下（“*用户*”为实际使用的用户名）。

[plugin@localhost examples]# vi collect\_ibmc\_logs.yml   
   
---   
- hosts: myhosts   
 connection: local   
 name: collect logs   
 gather\_facts: False   
   
# save\_mode: place to save logs: local, sftp, https, nfs, cifs, scp   
# file\_server\_ip: ip address of file server, if save\_mode is local, this parameter can be left blank.   
# file\_server\_user: the user of file server   
# file\_server\_pswd: the password of file server   
# file\_name: Log file storage path and file name   
   
 tasks:   
 - name: collect logs   
 ibmc\_collect\_logs :   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **save\_mode: "sftp"**   
 **file\_server\_ip: "sftp\_server\_ip"**   
 **file\_server\_user: "{{ sftp\_user }}"**   
 **file\_server\_pswd: "{{ sftp\_pswd }}"**   
 **file\_name: "/usr/dump.tar.gz"**

执行命令

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

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

1. 执行一键收集iBMC日志命令。

**ansible-playbook collect\_ibmc\_logs.yml**

如下返回信息表示命令执行成功。

[plugin@localhost examples]# ansible-playbook collect\_ibmc\_logs.yml   
/usr/local/python3/lib/python3.7/site-packages/cryptography-3.1.1-py3.7-linux-x86\_64.egg/cryptography/hazmat/bindings/openssl/binding.py:177: CryptographyDeprecationWarning: OpenSSL version 1.0.2 is no longer supported by the OpenSSL project, please upgrade. The next version of cryptography will drop support for it.   
 utils.CryptographyDeprecationWarning,   
   
PLAY [collect logs] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [collect logs] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

### 收集SEL日志

参数配置

修改“/home/plugin/ibmc\_ansible/examples/collect\_sel\_logs.yml”文件。

* save\_mode：表示在远程文件服务器存储或者本地存储。远程文件服务器存储取值为“sftp”、“https”、“nfs”、“cifs”或“scp”；本地存储取值为“local”。
* file\_server\_ip：远程文件服务器IP地址，当“save\_mode”设置为“sftp”、“https”、“nfs”、“cifs”或“scp”时，需要设置。
* file\_server\_user：远程文件服务器用户名，当“save\_mode”设置为“sftp”、“https”、“cifs”或“scp”时，需要设置。
* file\_server\_pswd：远程文件服务器密码，当“save\_mode”设置为“sftp”、“https”、“cifs”或“scp”时，需要设置。
* file\_name：日志文件保存的路径与文件名。只设置文件名时，日志文件默认保存在/home/*用户*/ansible\_ibmc/report/collect\_SEL\_log/路径下（“*用户*”为实际使用的用户名）。

[plugin@localhost examples]# vi collect\_sel\_logs.yml   
   
---   
- hosts: myhosts   
 connection: local   
 name: collect sel logs   
 gather\_facts: False   
   
# save\_mode: place to save logs: local, sftp, https, nfs, cifs, scp   
# file\_server\_ip: ip address of file server, if save\_mode is local, this parameter can be left blank.   
# file\_server\_user: the user of file server   
# file\_server\_pswd: the password of file server   
# file\_name: Log file storage path and file name   
   
 tasks:   
 - name: collect sel logs   
 ibmc\_collect\_sel\_logs :   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **save\_mode: "local"**   
 **file\_server\_ip: "sftp\_server\_ip"**   
 **file\_server\_user: "{{ sftp\_user }}"**   
 **file\_server\_pswd: "{{ sftp\_pswd }}"**   
 **file\_name: "/home/plugin/SEL\_log.tar.gz"**

执行命令

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

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

1. 执行收集SEL日志命令。

**ansible-playbook collect\_sel\_logs.yml**

如下返回信息表示命令执行成功。

[plugin@localhost examples]# ansible-playbook collect\_sel\_logs.yml   
/usr/local/python3/lib/python3.7/site-packages/cryptography-3.1.1-py3.7-linux-x86\_64.egg/cryptography/hazmat/bindings/openssl/binding.py:177: CryptographyDeprecationWarning: OpenSSL version 1.0.2 is no longer supported by the OpenSSL project, please upgrade. The next version of cryptography will drop support for it.   
 utils.CryptographyDeprecationWarning,   
   
PLAY [collect sel logs] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [collect sel logs] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

### 清空SEL日志

操作步骤

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

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

1. 执行清空SEL日志命令。

**ansible-playbook clear\_sel\_logs.yml**

如下返回信息表示命令执行成功。

[plugin@localhost examples]# ansible-playbook clear\_sel\_logs.yml   
/usr/local/python3/lib/python3.7/site-packages/cryptography-3.1.1-py3.7-linux-x86\_64.egg/cryptography/hazmat/bindings/openssl/binding.py:177: CryptographyDeprecationWarning: OpenSSL version 1.0.2 is no longer supported by the OpenSSL project, please upgrade. The next version of cryptography will drop support for it.   
 utils.CryptographyDeprecationWarning,   
   
PLAY [clear sel logs] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [clear sel logs] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

## 通用接口

功能介绍

提供一个通用公共接口，通过配置它的URL地址和请求体，可实现Redfish的全部功能。

关于Redfish的更多内容请参见《[华为服务器 iBMC Redfish 接口说明](https://support.huawei.com/enterprise/zh/doc/EDOC1000126991)》。

参数配置

修改“/home/plugin/ibmc\_ansible/examples/common\_api.yml”文件。

* url：请求URL地址。路径需要以/redfish开头，如/redfish/v1/Chassis/1/Oem/Huawei/Actions/Chassis.ControlIndicatorLED。
* request\_method：请求方式。可取值为“patch”、“post”、“get”或“delete”。
* request\_body：请求消息体，请求体内容为json格式，如 **'{"IndicatorLED":"Blinking", "Duration":50}'**。当请求方式设置为“get”或“delete”时，不可配置该参数。

[plugin@localhost examples]# vi common\_api.yml   
   
---   
- hosts: myhosts   
 connection: local   
 name: common api   
 gather\_facts: False   
   
# url: request resource   
# request\_method: request method: GET, POST, PATCH, DELETE   
# request\_body: request body content   
   
 tasks:   
 - name: common api   
 ibmc\_common\_api:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **url: "/redfish/v1/Chassis/1/Oem/Huawei/Actions/Chassis.ControlIndicatorLED"**   
 **request\_method: "POST"**   
 **request\_body: '{"IndicatorLED":"Blinking", "Duration":50}'**

执行命令

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

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

1. 执行配置命令。

**ansible-playbook common\_api.yml**

如下返回信息表示命令执行成功。

[plugin@localhost examples]# ansible-playbook common\_api.yml   
/usr/local/python3/lib/python3.7/site-packages/cryptography-3.1.1-py3.7-linux-x86\_64.egg/cryptography/hazmat/bindings/openssl/binding.py:177: CryptographyDeprecationWarning: OpenSSL version 1.0.2 is no longer supported by the OpenSSL project, please upgrade. The next version of cryptography will drop support for it.   
 utils.CryptographyDeprecationWarning,   
   
PLAY [common api] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [common api] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

## 文件本地传输

功能介绍

支持上传本地文件、下载文件至本地。

### 上传本地文件

参数配置

修改“/home/plugin/ibmc\_ansible/examples/upload\_file.yml”文件的“imgfile”参数。

imgfile：待上传本地文件的所在路径以及文件名，输入形式为“文件路径/文件名”，如/home/plugin/SOO.keytab。上传成功后的文件保存在iBMC的/tmp/web目录下。

* x86 V3服务器支持的文件类型如下：

hpm、cer、pem、cert、crt、pfx、p12、xml、keys、pub

* x86 V5服务器支持的文件类型如下：

hpm、zip、asc、cer、pem、cert、crt、pfx、p12、xml、keys、pub、keytab

* x86 V3服务器hpm文件最大允许46M；x86 V5服务器hpm、zip、asc文件最大允许90M。
* cer、pem、cert、crt、xml、p12、keytab文件最大允许1M；pfx、keys文件最大允许2M，pub文件最大允许2KB。

[plugin@localhost examples]# vi upload\_file.yml   
   
---   
- hosts: myhosts   
 connection: local   
 name: file upload   
 gather\_facts: False   
   
# imgfile: User-specified file to be uploaded, The format is file\_path/file\_name. After the upload is successful, the file is placed in the /tmp/web on iBMC.   
 # The file types allowed by the V3 board are as follows: {"hpm","cer","pem","cert","crt","pfx","p12","xml","keys","pub"}   
 # The file types allowed by the V5 board are as follows: {"hpm","zip","asc","cer","pem","cert","crt","pfx","p12","xml","keys","pub"," keytab"}   
 # The maximum allowable hpm file of V3 single-board is 46M; the maximum allowable hpm, zip, asc file of v5 single-board is 90M.   
 # The maximum allowable size of cer, pem, cert, crt, xml, p12, and keytab files is 1M.   
 # The maximum allowable size of pfx and keys files is 2M, and the maximum allowable size of pub files is 2KB.   
   
 tasks:   
 - name: file upload   
 ibmc\_upload\_file :   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **imgfile: "/home/plugin/SOO.keytab"**

执行命令

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

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

1. 执行上传本地文件命令。

**ansible-playbook upload\_file.yml**

如下返回信息表示命令执行成功。

[plugin@localhost examples]# ansible-playbook upload\_file.yml   
/usr/local/python3/lib/python3.7/site-packages/cryptography-3.1.1-py3.7-linux-x86\_64.egg/cryptography/hazmat/bindings/openssl/binding.py:177: CryptographyDeprecationWarning: OpenSSL version 1.0.2 is no longer supported by the OpenSSL project, please upgrade. The next version of cryptography will drop support for it.   
 utils.CryptographyDeprecationWarning,   
   
PLAY [file upload] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [file upload] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

### 下载文件至本地

参数配置

修改“/home/plugin/ibmc\_ansible/examples/download\_file.yml”文件中的“file\_name”和“local\_path”参数。

* file\_name：需要下载的iBMC上的文件，只能是/tmp/web目录或者/tmp/web子目录下的文件。

例如：如果文件路径为/tmp/web/111.txt，此处的配置应为**111.txt**；如果文件路径为/tmp/web/\*\*\*/111.txt，此处的配置应为**/\*\*\*/111.txt**。

* local\_path：文件的本地保存路径，如果不配置，默认下载到/home/*用户*/ansible\_ibmc/report/download/目录下。



* *用户*：表示实际使用的用户名。
* 下载后文件的名称会自动修改，会在原来的名称前面加上IP和时间。

[plugin@localhost examples]# vi download\_file.yml   
   
---   
- hosts: myhosts   
 connection: local   
 name: file download   
 gather\_facts: False   
   
# file\_name: the name of the file to be downloaded, from /tmp/web of iBMC   
# local\_path: local path for storing files, The default file save path is /home/USER/ansible\_ibmc/report/download/   
   
 tasks:   
 - name: file download   
 ibmc\_download\_file :   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 **file\_name: "SOO.keytab"**   
 **local\_path: "/home/plugin/"**

执行命令

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

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

1. 执行下载文件命令。

**ansible-playbook download\_file.yml**

如下返回信息表示命令执行成功。

[plugin@localhost examples]# ansible-playbook download\_file.yml   
/usr/local/python3/lib/python3.7/site-packages/cryptography-3.1.1-py3.7-linux-x86\_64.egg/cryptography/hazmat/bindings/openssl/binding.py:177: CryptographyDeprecationWarning: OpenSSL version 1.0.2 is no longer supported by the OpenSSL project, please upgrade. The next version of cryptography will drop support for it.   
 utils.CryptographyDeprecationWarning,   
   
PLAY [file download] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [file download] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

1. FAQ

* 1. 如何加密文件以及如何查看、编辑和执行已加密的文件

在使用加密命令（**ansible-vault**）之前，可通过执行**ansible-vault -h**查看该命令的相关帮助信息。

[root@localhost ~]# ansible-vault -h   
Usage: ansible-vault [create|decrypt|edit|encrypt|encrypt\_string|rekey|view] [options] [vaultfile.yml]   
   
encryption/decryption utility for Ansible data files   
   
Options:   
 --ask-vault-pass ask for vault password   
 -h, --help show this help message and exit   
 --new-vault-id=NEW\_VAULT\_ID   
 the new vault identity to use for rekey   
 --new-vault-password-file=NEW\_VAULT\_PASSWORD\_FILE   
 new vault password file for rekey   
 --vault-id=VAULT\_IDS the vault identity to use   
 --vault-password-file=VAULT\_PASSWORD\_FILES   
 vault password file   
 -v, --verbose verbose mode (-vvv for more, -vvvv to enable   
 connection debugging)   
 --version show program's version number, config file location,   
 configured module search path, module location,   
 executable location and exit   
   
 See 'ansible-vault <command> --help' for more information on a specific   
command.

* + 1. 如何加密文件

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

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

执行以下命令加密文件。

**ansible-vault encrypt** *文件名称*

如加密get\_basic\_info.yml文件：

执行**ansible-vault encrypt get\_basic\_info.yml**命令，提示输入密码。

[plugin@localhost examples]# ansible-vault encrypt get\_basic\_info.yml   
New Vault password:

输入需设置的密码，按“Enter”，提示再次输入密码。

[plugin@localhost examples]# ansible-vault encrypt get\_basic\_info.yml   
New Vault password:   
Confirm New Vault password:

再次输入设置的密码，按“Enter”，显示文件加密成功。

[plugin@localhost examples]# ansible-vault encrypt get\_basic\_info.yml   
New Vault password:   
Confirm New Vault password:   
Encryption successful

----结束

* + 1. 如何查看已加密文件

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

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

执行以下命令查看加密文件。

**ansible-vault view** *文件名称*

如查看已加密的get\_basic\_info.yml文件：

执行**ansible-vault view get\_basic\_info.yml**命令，提示输入文件的密码。

[plugin@localhost examples]# ansible-vault view get\_basic\_info.yml   
Vault password:

输入文件的密码，查看文件内容。

[plugin@localhost examples]# ansible-vault view get\_basic\_info.yml   
Vault password:   
---   
- hosts: myhosts   
 connection: local   
 name: get bmc basic info   
 gather\_facts: False   
   
# cvs\_format: Whether to write the result to a CSV file. It is a mandatory parameter. Available values: True, False   
   
 tasks:   
 - name: get bmc basic info   
 ibmc\_get\_basic\_info:   
 ibmc\_ip: "{{ ibmc\_ip }}"   
 ibmc\_user: "{{ ibmc\_user }}"   
 ibmc\_pswd: "{{ ibmc\_pswd }}"   
 csv\_format: True

----结束

* + 1. 如何编辑已加密文件

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

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

执行以下命令编辑加密文件。

**ansible-vault edit** *文件名称*

如编辑已加密的get\_basic\_info.yml文件：

执行**ansible-vault edit get\_basic\_info.yml**命令，提示输入文件的密码。

[plugin@localhost examples]# ansible-vault edit get\_basic\_info.yml   
Vault password:

输入文件的密码，编辑文件内容。

----结束

* + 1. 文件加密后，如何执行配置命令

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

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

执行配置命令。

**ansible-playbook -vv** *文件名称* **--ask-vault-pass**

如查询服务器基本信息：

执行**ansible-playbook -vv get\_basic\_info.yml --ask-vault-pass**命令，提示输入文件的密码，此处以Python3环境为例。

[plugin@localhost examples]# ansible-playbook -vv get\_basic\_info.yml --ask-vault-pass   
/usr/local/python3/lib/python3.7/site-packages/cryptography-3.1.1-py3.7-linux-x86\_64.egg/cryptography/hazmat/bindings/openssl/binding.py:177: CryptographyDeprecationWarning: OpenSSL version 1.0.2 is no longer supported by the OpenSSL project, please upgrade. The next version of cryptography will drop support for it.   
 utils.CryptographyDeprecationWarning,   
ansible-playbook 2.9.9   
 config file = None   
 configured module search path = ['/root/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']   
 ansible python module location = /usr/local/python3/lib/python3.7/site-packages/ansible-2.9.9-py3.7.egg/ansible   
 executable location = /usr/local/python3/bin/ansible-playbook   
 python version = 3.7.5 (default, Nov 16 2020, 23:36:26) [GCC 4.8.5 20150623 (Red Hat 4.8.5-44)]   
No config file found; using defaults   
Vault password:

输入文件的密码，查询服务器基本信息。

[plugin@localhost examples]# ansible-playbook -vv get\_basic\_info.yml --ask-vault-pass   
/usr/local/python3/lib/python3.7/site-packages/cryptography-3.1.1-py3.7-linux-x86\_64.egg/cryptography/hazmat/bindings/openssl/binding.py:177: CryptographyDeprecationWarning: OpenSSL version 1.0.2 is no longer supported by the OpenSSL project, please upgrade. The next version of cryptography will drop support for it.   
 utils.CryptographyDeprecationWarning,   
ansible-playbook 2.9.9   
 config file = None   
 configured module search path = ['/root/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']   
 ansible python module location = /usr/local/python3/lib/python3.7/site-packages/ansible-2.9.9-py3.7.egg/ansible   
 executable location = /usr/local/python3/bin/ansible-playbook   
 python version = 3.7.5 (default, Nov 16 2020, 23:36:26) [GCC 4.8.5 20150623 (Red Hat 4.8.5-44)]   
No config file found; using defaults   
Vault password:   
   
PLAYBOOK: get\_basic\_info.yml \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
1 plays in get\_basic\_info.yml   
   
PLAY [get bmc basic info] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
META: ran handlers   
   
TASK [get bmc basic info] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
task path: /home/ibmc\_ansible/examples/get\_basic\_info.yml:10   
ok: [host0] => {"ansible\_facts": {"discovered\_interpreter\_python": "/usr/bin/python"}, "changed": false, "msg": "Get basic info successful! For more detail information,please refer the report log: /home/root/ansible\_ibmc/report/basic\_info/192.168.2.10\_BasicInfo.json"}   
META: ran handlers   
META: ran handlers   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0 : ok=1 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

----结束

1. 获取技术支持

如果在设备维护或故障处理过程中，遇到难以确定或难以解决的问题，通过文档的指导仍然不能解决，请通过如下方式获取技术支持：

* 联系客户服务中心。
* 中国区企业用户请通过以下方式联系我们：
  1. 客户服务电话：400-822-9999
  2. 客户服务邮箱：[support\_e@huawei.com](mailto:support_e@huawei.com)
* 企业网全球各地区客户服务热线可以通过以下网站查找：[企业用户全球服务热线](https://e.huawei.com/cn/service-hotline-query)。
* 中国区运营商用户请通过以下方式联系我们：
  1. 客户服务电话：400-830-2118
  2. 客户服务邮箱：[support@huawei.com](mailto:support@huawei.com)
* 运营商全球各地区客户服务热线可以通过以下网站查找：[运营商用户全球服务热线](https://support.huawei.com/carrier/docview!docview?nid=NEWS1000000994)。
* 联系当地技术支持人员。