|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | | |  |
| **Huawei Ansible Plug-in V1.3.2** | | |  |
| **用户指南** | | |
| **文档版本** | **03** | |
| **发布日期** | **2019-09-25** | |
|  | | | | |
|  | 华为技术有限公司 | |  |  |

|  |
| --- |
| 版权所有 © 华为技术有限公司2019。 保留一切权利。  非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。  商标声明  和其他华为商标均为华为技术有限公司的商标。  本文档提及的其他所有商标或注册商标，由各自的所有人拥有。  注意  您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或默示的声明或保证。  由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。 |

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

前言

概述

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

读者对象

本指南主要适用于以下工程师：

* 技术支持工程师
* 系统维护工程师

符号约定

在本文中可能出现下列标志，它们所代表的含义如下。

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

修改记录

| **文档版本** | **发布日期** | **修改说明** |
| --- | --- | --- |
| 03 | 2019-09-25 | 新增4.5 管理iBMC用户。 |
| 02 | 2018-09-11 | 修改4.11 RAID配置。 |
| 01 | 2018-04-26 | 第一次正式发布。 |

目 录

[前言 ii](#_Toc20757090)

[1 简介 1](#_Toc20757091)

[2 安装和卸载华为Ansible插件 2](#_Toc20757092)

[2.1 环境依赖 2](#_Toc20757093)

[2.1.1 软件依赖 2](#_Toc20757094)

[2.1.2 硬件依赖 2](#_Toc20757095)

[2.1.3 版本配套关系 2](#_Toc20757096)

[2.2 安装华为Ansible插件 3](#_Toc20757097)

[2.3 卸载华为Ansible插件 3](#_Toc20757098)

[3 配置华为Ansible插件 4](#_Toc20757099)

[3.1 修改/etc/ansible/ansible.cfg文件 4](#_Toc20757100)

[3.2 配置/etc/ansible/hosts文件 4](#_Toc20757101)

[3.3 配置/group\_vars/myhosts文件 4](#_Toc20757102)

[4 使用华为Ansible插件 6](#_Toc20757103)

[4.1 查询信息（生成csv文件） 6](#_Toc20757104)

[4.2 查询健康状态（生成csv文件） 8](#_Toc20757105)

[4.3 配置启动设备 9](#_Toc20757106)

[4.4 管理电源 10](#_Toc20757107)

[4.5 管理iBMC用户 11](#_Toc20757108)

[4.5.1 查询iBMC用户（生成json文件） 11](#_Toc20757109)

[4.5.2 创建iBMC用户 12](#_Toc20757110)

[4.5.3 修改iBMC用户 14](#_Toc20757111)

[4.5.4 删除iBMC用户 15](#_Toc20757112)

[4.6 iBMC配置 16](#_Toc20757113)

[4.6.1 查询iBMC配置信息 16](#_Toc20757114)

[4.6.2 配置iBMC 17](#_Toc20757115)

[4.7 管理NTP服务 19](#_Toc20757116)

[4.7.1 查询NTP服务信息 19](#_Toc20757117)

[4.7.2 配置NTP 20](#_Toc20757118)

[4.8 管理SNMP服务 22](#_Toc20757119)

[4.8.1 查询SNMP服务信息（生成json文件） 22](#_Toc20757120)

[4.8.2 配置SNMP Trap 23](#_Toc20757121)

[4.9 导入导出Profile文件 25](#_Toc20757122)

[4.10 固件升级 26](#_Toc20757123)

[4.10.1 查询固件信息（生成json文件） 26](#_Toc20757124)

[4.10.2 升级固件 27](#_Toc20757125)

[4.10.2.1 BMC方式 27](#_Toc20757126)

[4.10.2.2 Smart Provisioning方式 29](#_Toc20757127)

[4.11 RAID配置 30](#_Toc20757128)

[4.11.1 查询RAID配置 30](#_Toc20757129)

[4.11.2 删除RAID组 32](#_Toc20757130)

[4.11.3 创建RAID组 34](#_Toc20757131)

[4.11.4 修改RAID配置 35](#_Toc20757132)

[4.12 OS部署 38](#_Toc20757133)

[4.12.1 ServiceCD2.0方式 38](#_Toc20757134)

[4.12.2 Smart Provisioning方式 41](#_Toc20757135)

[5 FAQ 45](#_Toc20757136)

[5.1 如何配置系统的多个用户均可以使用Ansible 45](#_Toc20757137)

# 简介

华为Ansible插件是一个集成在Ansible软件中的管理华为服务器的插件，通过Redfish接口，该插件可以实现对华为服务器配置、部署、升级的功能。

通过该插件，可以实现以下功能：

* 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的信息。
* 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。
* 升级服务器的固件（支持BMC或Smart Provisioning方式）。
* 部署服务器的OS（支持ServiceCD2.0或Smart Provisioning方式）。

# 安装和卸载华为Ansible插件

[2.1 环境依赖](#_ZH-CN_TOPIC_0102786727)

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

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

## 环境依赖

### 软件依赖

* sshpass 1.06及以后版本
* Python 2.7和Python-request 2.6.0及以后版本
* jq 1.5及以后版本
* 文件服务器，需要支持CIFS或NFS协议

### 硬件依赖

* 支持单RAID卡的RAID配置：支持带外管理，如LSI SAS3108等（目前仅测试LSI SAS3108 RAID卡）。
* 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。

### 版本配套关系

* 仅支持2.0.0及以上版本的Ansible。
* 仅支持0.51及以上版本的BIOS。
* 仅支持3.20及以上版本的iBMC。
* 仅支持V116及以上版本的Smart Provisioning。
* 仅支持V133及以后版本的ServiceCD2.0，可在[Support E](http://support.huawei.com/enterprise/zh/server/fusionserver-tools-pid-21015513/software)网站下载。
* 仅支持V1.30.211及以后版本的iBMA（带内查询时需安装），可在[Support E](http://support.huawei.com/enterprise/zh/server/ibma-pid-21099187/software)网站下载。

## 安装华为Ansible插件

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

将安装包上传到Ansible服务器的任意目录（如“/home”目录）。

以root用户登录Ansible服务器。

进入华为Ansible插件的安装包所在目录。

执行以下命令，解压华为Ansible插件的软件包。

**tar -xvf Huawei\_iBMC\_Ansible\_Modules\_x.x.tar**

执行以下命令，进入解压后的“/Huawei-iBMC-Ansible\_Modules”目录。

**cd /Huawei-iBMC\_Ansible\_Modules**

执行以下命令，安装华为Ansible插件。

**python install.py**

----结束

## 卸载华为Ansible插件

以root用户登录Ansible服务器。

进入“/ansible\_ibmc”目录，执行卸载命令。

**cd /etc/ansible/ansible\_ibmc**

**python uninstall.py**

----结束

# 配置华为Ansible插件

[3.1 修改/etc/ansible/ansible.cfg文件](#_ZH-CN_TOPIC_0102786569)

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

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

## 修改/etc/ansible/ansible.cfg文件

[defaults]   
# some basic default values...   
#inventory = /etc/ansible/hosts   
library = /etc/ansible/ansible\_ibmc/module/

## 配置/etc/ansible/hosts文件

[myhosts]   
host0.domain.com ibmcip=192.168.2.20 host=huaweiserver0   
host1.domain.com ibmcip=192.168.2.21 host=huaweiserver1

## 配置/group\_vars/myhosts文件

设置“/etc/ansible/ansible\_ibmc/playbooks/group\_vars/”目录下的“myhosts”文件中的相关参数，如日志存储路径、iBMC用户名密码、SFTP/CIFS服务用户名密码等。

[root@ansible group\_vars]# cd /etc/ansible/ansible\_ibmc/playbooks/group\_vars/   
[root@ansible group\_vars]# vim 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   
   
# input local directory where all results are placed   
rootdir: /etc/ansible/ansible\_ibmc/report   
reportdir: /etc/ansible/ansible\_ibmc/report   
   
# input True if we want the inventory query result to be comma-separated   
csvformat: True   
   
# input the huawei ibmc user and password   
ibmcuser: "ibmc\_user"   
ibmcpswd: "ibmc\_pwd"   
   
# input the scp user and password when we need to use the scp service   
ibmcuser: "scp\_user"   
ibmcpswd: "scp\_pwd"   
   
# input the sftp user and password when we need to use the sftp service   
sftpuser: "sftp\_user"   
sftppswd: "sftp\_pwd"   
   
# input the cifs user and password when we need to use the cifs service   
cifsuser: "cifs\_user"   
cifspswd: "cifs\_pwd"   
   
   
# if you select SNMP Trap mode as V1 or V2C, you need to set the community name   
community: "community\_name"   
   
 ~



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

# 使用华为Ansible插件

[4.1 查询信息（生成csv文件）](#_ZH-CN_TOPIC_0102786712)

[4.2 查询健康状态（生成csv文件）](#_ZH-CN_TOPIC_0102786717)

[4.3 配置启动设备](#_ZH-CN_TOPIC_0102786708)

[4.4 管理电源](#_ZH-CN_TOPIC_0102786699)

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

[4.6 iBMC配置](#_ZH-CN_TOPIC_0193957036)

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

[4.8 管理SNMP服务](#_ZH-CN_TOPIC_0193957039)

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

[4.10 固件升级](#_ZH-CN_TOPIC_0102786721)

[4.11 RAID配置](#_ZH-CN_TOPIC_0102786725)

[4.12 OS部署](#_ZH-CN_TOPIC_0102786703)

## 查询信息（生成csv文件）

功能介绍

支持查询服务器状态、服务器型号、BIOS版本、CPLD版本、RAID卡版本、网卡版本、资产标签、序列号、内存大小、CPU型号、CPU个数、电源状态、功耗统计、BMC版本和启动设备。



查询RAID卡和网卡版本需要安装V1.30.211及以后版本的iBMA，可在[Support E](http://support.huawei.com/enterprise/zh/server/ibma-pid-21099187/software)网站下载。

参数配置

待查询的服务器信息的字段已全部配置到“inventory.yml”文件中，查询前用户无需修改。

查询信息后生成的csv文件默认保存在“/etc/ansible/ansible\_ibmc/report”目录中，建议导出csv文件后再查看。

执行操作

进入“playbooks”文件目录。

**cd /etc/ansible/ansible\_ibmc/playbooks**

执行查询命令。

root@ubuntu:/etc/ansible/ansible\_ibmc/playbooks# ansible-playbook inventory.yml   
   
TASK [Add header row to system information CSV file] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
changed: [host3.domain.com -> localhost] => (item=ServerStatus)   
changed: [host3.domain.com -> localhost] => (item=ServerModel)   
changed: [host3.domain.com -> localhost] => (item=BiosVersion)   
changed: [host3.domain.com -> localhost] => (item=AssetTag)   
changed: [host3.domain.com -> localhost] => (item=SerialNumber)   
changed: [host3.domain.com -> localhost] => (item=MemoryGiB)   
changed: [host3.domain.com -> localhost] => (item=MemoryHealth)   
changed: [host3.domain.com -> localhost] => (item=CPUModel)   
changed: [host3.domain.com -> localhost] => (item=CPUHealth)   
changed: [host3.domain.com -> localhost] => (item=CPUCount)   
changed: [host3.domain.com -> localhost] => (item=PowerState)   
changed: [host3.domain.com -> localhost] => (item=ConsumedWatts)   
changed: [host3.domain.com -> localhost] => (item=iBMCFirmwareVersion)   
changed: [host3.domain.com -> localhost] => (item=iBMCHealth)   
changed: [host3.domain.com -> localhost] => (item=BootSourceOverrideMode)   
   
TASK [command] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
changed: [host3.domain.com -> localhost]   
   
TASK [Combine all host outputs to one CSV file] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
changed: [host3.domain.com -> localhost]   
   
TASK [Remove temporary results file] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
changed: [host3.domain.com -> localhost]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host3.domain.com : ok=88 changed=38 unreachable=0 failed=0

查询信息后生成的csv文件（如“allhosts\_inventory\_20180507\_220419.csv”）默认保存在“/etc/ansible/ansible\_ibmc/report”目录中，建议导出csv文件后再查看，如图4-1所示。

查询结果



----结束

## 查询健康状态（生成csv文件）

功能介绍

支持查询服务器整机、风扇、内存、BMC、电源、硬盘和CPU的健康状态。

参数配置

待查询的服务器部件健康状态的字段已全部配置到“healthStatus.yml”文件中，查询前用户无需修改。

查询服务器部件状态后生成的csv文件默认保存在“/etc/ansible/ansible\_ibmc/playbooks/report”目录中，建议导出csv文件后再查看。

执行操作

执行查询命令。

root@ubuntu:/etc/ansible/ansible\_ibmc/playbooks# ansible-playbook healthStatus.yml   
   
TASK [Add header row to system information CSV file] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
changed: [host7.domain.com -> localhost] => (item=ServerStatus)   
changed: [host7.domain.com -> localhost] => (item=CPUHealth)   
changed: [host7.domain.com -> localhost] => (item=MemoryHealth)   
changed: [host7.domain.com -> localhost] => (item=iBMCHealth)   
changed: [host7.domain.com -> localhost] => (item=FansHealth)   
changed: [host7.domain.com -> localhost] => (item=PowerSuppliesHealth)   
changed: [host7.domain.com -> localhost] => (item=DiskHealth)   
   
TASK [command] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
changed: [host7.domain.com -> localhost]   
TASK [Combine all host outputs to one CSV file] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
changed: [host7.domain.com -> localhost]   
TASK [Remove temporary results file] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
changed: [host7.domain.com -> localhost]   
TASK [Remove temporary results file] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host7.domain.com -> localhost]   
TASK [command] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
skipping: [host7.domain.com]   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host7.domain.com : ok=48 changed=22 unreachable=0 failed=0

查询服务器部件状态后生成的csv文件（如“allhosts\_health\_20180507\_223830.csv”）默认保存在“/etc/ansible/ansible\_ibmc/report”目录中，建议导出csv文件后再查看，如图4-2所示。

查询结果



## 配置启动设备

功能介绍

支持设置启动设备为PXE、Cd、Hdd、Floppy或BiosSetup，生效方式默认设置为单次生效。

参数配置

修改“/etc/ansible/ansible\_ibmc/setBootDevice.yml”文件，包括“hosts”（配置对象）和“command”（可设置为“Cd”、“Pxe”、“Floppy”、“Hdd”、“BiosSetup”或“None”）。

- hosts: myhosts   
 name: set boot device   
 gather\_facts: False   
 # boot device:   
 # Cd CD ROM   
 # Pxe pxe   
 # Floppy Floppy   
 # Hdd hard disk device   
 # BiosSetup   
 # None   
 tasks:   
 - name: set boot order   
 local\_action: >   
 ibmc category=SetBootDevice command='Cd' ibmcip={{ ibmcip }}   
 ibmcuser={{ ibmcuser }} ibmcpswd={{ ibmcpswd }}

执行配置

执行配置命令。

[root@localhost playbooks]# ansible-playbook setBootDevice.yml   
PLAY [set boot device] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
TASK [set boot order] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host7.domain.com -> localhost]   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host7.domain.com : ok=1 changed=0 unreachable=0 failed=0

查看历史配置记录。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/report/cfgBmc.log   
[2018-01-25 06:00:12 INFO ] (cfgBmc.py:186)- 192.168.3.114 -- config bmc successful!   
[2018-01-25 06:11:10 INFO ] (cfgBmc.py:101)- 192.168.3.114 -- set boot device as:Hdd successful!

## 管理电源

功能介绍

支持上电、下电、强制重启和正常关机。

参数配置

修改“/etc/ ansible/ansible\_ibmc/playbooks/powerManage.yml”文件，包括“hosts”（配置对象）和“command”参数，“command”可设置为“PowerOn”（上电）、“PowerOff”（下电）、“ForceRestart”（强制重启）、“GracefulShutdown”（正常关机）、

“ForcePowerCycle”（强制下电重启）或Nmi。

- hosts: myhosts   
 name: iBMC Manage System Power   
 gather\_facts: False   
   
# Commands available:   
# PowerOn Turn system power on   
# PowerOff Turn system power off, please confirm it again before run the command!   
# ForceRestart Reset OS gracefully, is invalid for server in power-off state   
# GracefulShutdown Shutdown OS gracefully   
# ForcePowerCycle Forcibly power off and then power on   
# Nmi generate a non-maskable interrupt only when the OS is abnormal   
   
   
 tasks:   
   
 # Except for PowerSate, tasks are mutually exclusive, do only one at a time   
   
 - name: Turn system power on   
 local\_action: >   
 ibmc category=Power command=PowerOn ibmcip={{ ibmcip }}   
 ibmcuser={{ ibmcuser }} ibmcpswd={{ ibmcpswd }}

执行配置

执行电源管理命令。

[root@localhost playbooks]# ansible-playbook powerManage.yml   
PLAY [iBMC Manage System Power] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
TASK [Force restart] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host7.domain.com -> localhost]   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host7.domain.com : ok=1 changed=0 unreachable=0 failed=0

查看历史操作记录。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/report/pmReport.log   
[2018-01-25 06:03:23 INFO ] (powerManage.py:120)- 192.168.3.114 -- set system force restart successful!

## 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）

参数配置

查询iBMC用户信息后自动生成的json文件默认保存在“/etc/ansible/ansible\_ibmc/report”目录中，建议导出json文件后再查看。

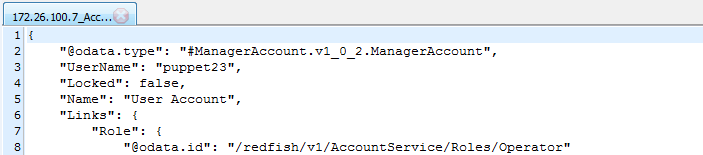
执行操作

执行查询命令。

[root@localhost playbooks]# ansible-playbook getAccount.yml   
   
PLAY [Get Account] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [Get Account] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com -> localhost]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0

查询iBMC用户信息后自动生成的json文件（如“172.26.100.7\_AccountInfo.json”）默认保存在“/etc/ansible/ansible\_ibmc/report”目录中，建议导出json文件后再查看，如图4-3所示。

查询结果



### 创建iBMC用户

参数配置

* 修改“/etc/ansible/ansible\_ibmc/playbooks/group\_vars/myhosts”文件，包括“newuser”（用户名）和“newpswd”（用户密码）参数。

# 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   
   
# input local directory where all results are placed   
rootdir: /etc/ansible/ansible\_ibmc/report   
reportdir: /etc/ansible/ansible\_ibmc/report   
   
# input True if we want the inventory query result to be comma-separated   
csvformat: True   
   
#used for create new ibmc account   
newuser: "newuser"   
newpswd: "newpswd"   
   
# input the huawei ibmc user and password   
ibmcuser: "ibmc\_user"   
ibmcpswd: "ibmc\_pwd"   
   
# input the sftp user and password when we need to use the sftp service   
sftpuser: "sftp\_user"   
sftppswd: "sftp\_pwd"   
   
# input the cifs user and password when we need to use the cifs service   
cifsuser: "cifs\_user"   
cifspswd: "cifs\_pwd"   
   
   
# if you select SNMP Trap mode as V1 or V2C, you need to set the community name   
community: "community\_name"   
# input the scp user and password when we need to use the scp service   
scpuser: "scp\_user"   
scppswd: "scp\_pwd"

* 修改“/etc/ansible/ansible\_ibmc/playbooks/createAccount.yml”文件，包括“hosts”（配置对象）和“roleid”（用户角色）参数。

关于参数的具体信息可参考[《华为服务器 iBMC Redfish 接口说明》](http://support.huawei.com/enterprise/zh/doc/EDOC1000126991)的“创建新用户”章节。

- hosts: myhosts   
 name: created Account   
 gather\_facts: no   
   
   
 tasks:   
   
   
 - name: create account   
 local\_action: >   
 ibmc category=ManageAccount command="createAccount" useraccount={{ newuser }} newpassword={{ newpswd }} roleid='Administrator' ibmcip={{ ibmcip }}   
 ibmcuser={{ ibmcuser }} ibmcpswd={{ ibmcpswd }}

执行操作

执行创建命令。

[root@localhost playbooks]# ansible-playbook createAccount.yml   
   
PLAY [created Account] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [create account] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com -> localhost]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0

查看历史创建记录。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/report/accountManage.log   
[2019-09-29 06:43:07 INFO ] (accountManage.py:194)- 172.26.100.7 -- create account successful!

### 修改iBMC用户

参数配置

修改“/etc/ansible/ansible\_ibmc/configFile/accountCfg/”目录下的“modifyAccount.json”文件。

关于参数的具体信息可参考[《华为服务器 iBMC Redfish 接口说明》](http://support.huawei.com/enterprise/zh/doc/EDOC1000126991)的“修改指定用户信息”章节。

{   
"test9": {   
"RoleId":"Operator",   
"Enabled":true,   
"Oem":{   
 "Huawei":{   
 "AccountInsecurePromptEnabled":false,   
 "LoginInterface":["Web","SNMP","IPMI","SSH","SFTP","Local","Redfish"]   
 }   
 }   
 }   
}

执行操作

执行修改命令。

[root@localhost playbooks]# ansible-playbook modifyAccount.yml   
   
PLAY [modify Account] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [modify Account] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com -> localhost]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0

查看历史修改记录。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/report/accountManage.log   
[2019-09-29 06:53:10 INFO ] (accountManage.py:247)- 172.26.100.7 -- modify account successful! respon json is :{u'@odata.type': u'#ManagerAccount.v1\_0\_2.ManagerAccount', u'UserName': u'test5', u'Locked': False, u'Name': u'User Account', u'Links': {u'Role': {u'@odata.id': u'/redfish/v1/AccountService/Roles/Operator'}}, u'RoleId': u'Operator', u'@Message.ExtendedInfo': [{u'Severity': u'Warning', u'MessageId': u'iBMC.1.0.AccountInsecurePromptModificationRestricted', u'RelatedProperties': [], u'Message': u'Only the account itself can modify the status of the account insecure prompt.', u'Resolution': u'None', u'MessageArgs': []}], u'Enabled': True, u'@odata.id': u'/redfish/v1/AccountService/Accounts/6', u'@odata.context': u'/redfish/v1/$metadata#AccountService/Accounts/Members/$entity', u'Oem': {u'Huawei': {u'LoginRule': [], u'PasswordValidityDays': None, u'SSHPublicKeyHash': None, u'LoginInterface': [u'Web', u'SNMP', u'IPMI', u'SSH', u'SFTP', u'Local', u'Redfish'], u'Deleteable': True, u'AccountInsecurePromptEnabled': True, u'Actions': {u'#Account.ImportSSHPublicKey': {u'target': u'/redfish/v1/AccountService/Accounts/6/Oem/Huawei/Actions/Account.ImportSSHPublicKey', u'@Redfish.ActionInfo': u'/redfish/v1/AccountService/Accounts/6/ImportSSHPublicKeyActionInfo'}, u'#Account.DeleteMutualAuthClientCert': {u'target': u'/redfish/v1/AccountService/Accounts/6/Oem/Huawei/Actions/Account.DeleteMutualAuthClientCert', u'@Redfish.ActionInfo': u'/redfish/v1/AccountService/Accounts/6/DeleteMutualAuthClientCertActionInfo'}, u'#Account.DeleteSSHPublicKey': {u'target': u'/redfish/v1/AccountService/Accounts/6/Oem/Huawei/Actions/Account.DeleteSSHPublicKey', u'@Redfish.ActionInfo': u'/redfish/v1/AccountService/Accounts/6/DeleteSSHPublicKeyActionInfo'}, u'#Account.ImportMutualAuthClientCert': {u'target': u'/redfish/v1/AccountService/Accounts/6/Oem/Huawei/Actions/Account.ImportMutualAuthClientCert', u'@Redfish.ActionInfo': u'/redfish/v1/AccountService/Accounts/6/ImportMutualAuthClientCertActionInfo'}}, u'MutualAuthClientCert': None}}, u'Password': None, u'Id': u'6'}

### 删除iBMC用户

参数配置

修改“/etc/ansible/ansible\_ibmc/playbooks/deleteAccount.yml”文件，包括“hosts”（配置对象）和“useraccount”（需删除的用户名）参数。

- hosts: myhosts   
 name: delete account   
 gather\_facts: no   
   
   
 tasks:   
   
   
 - name: delete account   
 local\_action: >   
 ibmc category=ManageAccount command="deleteAccount" useraccount="test" ibmcip={{ ibmcip }}   
 ibmcuser={{ ibmcuser }} ibmcpswd={{ ibmcpswd }}

执行操作

执行删除命令。

[root@localhost playbooks]# ansible-playbook deleteAccount.yml   
   
PLAY [delete account] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [delete account] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com -> localhost]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0

查看历史删除记录。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/report/accountManage.log   
[2019-09-29 06:55:50 INFO ] (accountManage.py:147)- 172.26.100.7 -- delete account successful!

## iBMC配置

功能介绍

* 支持查询iBMC配置信息。
* 支持配置单个服务器的iBMC IP地址以及批量配置多个服务器iBMC IP地址的子网掩码和网关。

### 查询iBMC配置信息

参数配置

将“/etc/ansible/ansible\_ibmc/playbooks/cfgBMCIP.yml”文件中的“command”参数修改为“getinfo”。

- hosts: myhosts   
 name: config bmc ip   
 gather\_facts: False   
   
 tasks:   
   
 # config ibmc ip including IP address;Subnet mask;Gateway, must be separated by semicolons(;)!   
   
 - name: config bmc ip   
 local\_action: >   
 ibmc category=ConfigBMC command='getinfo' ibmcip={{ ibmcip }}   
 ibmcuser={{ ibmcuser }} ibmcpswd={{ ibmcpswd }}

执行配置

执行查询命令。

[root@localhost playbooks]# ansible-playbook cfgBMCIP.yml   
   
PLAY [config bmc ip] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [config bmc ip] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com -> localhost]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0

查看iBMC配置的详细信息。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/report/cfgBmc.log   
[2019-09-25 16:11:22 INFO ] (cfgBmc.py:259)- 172.26.100.7 -- get bmc successful! bmc json is:{u'@odata.type': u'#EthernetInterface.v1\_2\_0.EthernetInterface', u'VLAN': {u'VLANEnable': False, u'VLANId': 0}, u'Name': u'Manager Ethernet Interface', u'IPv6StaticAddresses': [{u'PrefixLength': 64, u'Address': u'2017:5::4'}], u'HostName': u'server7', u'@odata.id': u'/redfish/v1/Managers/1/EthernetInterfaces/485702ab0d57', u'@odata.context': u'/redfish/v1/$metadata#Managers/Members/1/EthernetInterfaces/Members/$entity', u'FQDN': u'server7.', u'NameServers': [u'', u''], u'IPv4Addresses': [{u'SubnetMask': u'255.255.0.0', u'AddressOrigin': u'Static', u'Gateway': u'172.26.0.1', u'Address': u'172.26.100.7'}], u'Link': {u'Oem': {u'Huawei': {u'EthernetInterface': {u'@odata.id': u'/redfish/v1/Systems/1/EthernetInterfaces'}}}}, u'IPv6DefaultGateway': u'2017:5::1', u'PermanentMACAddress': u'48:57:02:ab:0d:57', u'IPv6Addresses': [{u'AddressOrigin': u'Static', u'PrefixLength': 64, u'Address': u'2017:5::4'}, {u'AddressOrigin': u'LinkLocal', u'PrefixLength': 64, u'Address': u'fe80::4a57:2ff:feab:d57'}], u'Id': u'485702ab0d57', u'Oem': {u'Huawei': {u'ManagementNetworkPort': {u'PortNumber': 1, u'Type': u'Dedicated'}, u'NetworkPortMode': u'Fixed', u'DNSAddressOrigin': u'Static', u'IPVersion': u'IPv4AndIPv6', u'ManagementNetworkPort@Redfish.AllowableValues': [{u'PortNumber': 1, u'Type': u'Dedicated', u'LinkStatus': u'Connected'}, {u'PortNumber': 1, u'Type': u'LOM', u'LinkStatus': u'Connected'}, {u'PortNumber': 2, u'Type': u'LOM', u'LinkStatus': u'Disconnected'}, {u'PortNumber': 3, u'Type': u'LOM', u'LinkStatus': u'Disconnected'}, {u'PortNumber': 4, u'Type': u'LOM', u'LinkStatus': u'Disconnected'}], u'AdaptivePort': []}}}

### 配置iBMC

参数配置

1. 配置“/etc/ansible/ansible\_ibmc/configFile/ibmcCfg/ibmcIPcfg.json”文件中的相关参数，如下所示。

“AddressOrigin”可设置为“Static”或“DHCPv6”；“IPVersion”可设置为“IPv4”、“IPv6”或“IPv4AndIPv6”。

关于参数的具体信息可参考[《华为服务器 iBMC Redfish 接口说明》](http://support.huawei.com/enterprise/zh/doc/EDOC1000126991)的“修改指定iBMC网口信息”章节。

{   
 "IPv4Addresses" :   
 [   
 {   
 "Address":"172.26.100.10",   
 "SubnetMask":"255.255.0.0",   
 "AddressOrigin":"Static",   
 "Gateway":"172.26.0.1"   
 }   
 ],   
 "IPv6Addresses":[   
 {   
 "Address":"2017:5::9",   
 "PrefixLength":"32",   
 "AddressOrigin":"Static"   
 }   
   
 ],   
 "IPv6DefaultGateway": "2017:5::1",   
 "IPVersion": "IPv4AndIPv6"   
   
}

1. 修改“/etc/ansible/ansible\_ibmc/playbooks/cfgBMCIP.yml”文件，包括“hosts”（配置对象）和“command”（“ibmcIPcfg.json”文件所在的路径）参数。

- hosts: myhosts   
 name: config bmc   
 gather\_facts: False   
   
 tasks:   
   
 # Except for PowerSate, tasks are mutually exclusive, do only one at a time   
   
 - name: config bmc ip   
 local\_action: >   
 ibmc category=ConfigBMC command='/etc/ansible/ansible\_ibmc/configFile/ibmcCfg/ibmcIPcfg.json' ibmcip={{ ibmcip }}   
 ibmcuser={{ ibmcuser }} ibmcpswd={{ ibmcpswd }}

执行配置

执行配置命令。

[root@localhost playbooks]# ansible-playbook cfgBMCIP.yml   
PLAY [config bmc] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
TASK [config bmc ip] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host7.domain.com -> localhost]   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host7.domain.com : ok=1 changed=0 unreachable=0 failed=0

查看历史配置记录。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/report/cfgBmc.log   
[2018-01-25 06:00:12 INFO ] (cfgBmc.py:186)- 192.168.3.114 -- config bmc successful!

## 管理NTP服务

功能介绍

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

### 查询NTP服务信息

参数配置

将“/etc/ansible/ansible\_ibmc/playbooks/cfgNTP.yml”文件中的“command”参数修改为“getinfo”。

- hosts: myhosts   
 name: config NTP   
 gather\_facts: False   
   
 tasks:   
   
 # Config NTP function as below:   
 #   
 # 1,"ServiceEnabled": ServiceEnabled, true/false   
 # 2,"AlternateNtpServer": AlternateNtpServer, ip/domain name   
 # 3,"PreferredNtpServer": PreferredNtpServer, ip/domain name   
 # 4,"ServerAuthenticationEnabled": ServerAuthenticationEnabled, true/false   
 # 5,"NtpAddressOrigin": NtpAddressOrigin, IPv4/IPv6/Static   
 # 6,"MinPollingInterval": MinValue, 3~17 and less than MaxValue   
 # 7,"MaxPollingInterval": MaxValue, 3~17 and bigger than MinValue   
 #   
   
 - name: config bmc ntp   
 local\_action: >   
 ibmc category=ConfigNTP command='getinfo' ibmcip={{ ibmcip }}   
 ibmcuser={{ ibmcuser }} ibmcpswd={{ ibmcpswd }}

执行配置

执行查询命令。

[root@localhost playbooks]# ansible-playbook cfgNTP.yml   
   
PLAY [config NTP] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [config bmc ntp] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com -> localhost]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0

查看NTP服务详细信息。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/report/cfgNTP.log   
[2019-09-25 16:41:35 INFO ] (cfgNTP.py:189)- 172.26.100.7 -- get NTP successful! NTPInfo is:ServiceEnabled: False PreferredNtpServer:172.26.100.101 AlternateNtpServer:172.26.100.100 NtpAddressOrigin:Static MinPollingInterval:5 MaxPollingInterval:10 ServerAuthenticationEnabled:False NTPKeyStatus:Uploaded

### 配置NTP

参数配置

1. 需要配置“/etc/ansible/ansible\_ibmc/configFile/ntpCfg/ntp.ini”文件中的相关配置，如下所示。

关于参数的具体信息可参考[《华为服务器 iBMC Redfish 接口说明》](http://support.huawei.com/enterprise/zh/doc/EDOC1000126991)的“修改NTP资源属性”章节。

[NTP]   
# This is an ini file for bmc ntp config.   
   
# 1,"ServiceEnabled": ServiceEnabled, true/false   
# 2,"AlternateNtpServer": AlternateNtpServer, ip/domain name   
# 3,"PreferredNtpServer": PreferredNtpServer, ip/domain name   
# 4,"ServerAuthenticationEnabled": ServerAuthenticationEnabled, true/false   
# 5,"NtpAddressOrigin": NtpAddressOrigin, IPv4/IPv6/Static   
# 6,"MinPollingInterval": MinValue, 3~17 and <= MaxValue   
# 7,"MaxPollingInterval": MaxValue, 3~17 and >= MinValue   
   
   
# Enable or disable bmc ntp service,you can select true or false.   
ServiceEnabled = true   
# Config alternate NtpServer, you can enter ip or domain name.   
AlternateNtpServer = 192.168.100.2   
# Config preferred NtpServer, you can enter ip or domain name.   
PreferredNtpServer = 192.168.100.1   
# Enable or disable Server Authentication service,you can select true or false.   
ServerAuthenticationEnabled = false   
# Config Ntp Address Origin, you can select from IPv4/IPv6/Static.   
NtpAddressOrigin = IPv4   
# Config Min Polling Interval time, in 3~17 and <= MaxValue   
MinPollingInterval = 3   
# Config Max Polling Interval time, in 3~17 and >= MinValue   
MaxPollingInterval = 17

1. 修改“/etc/ansible/ansible\_ibmc/playbooks/cfgNTP.yml”文件，包括“hosts”（配置对象）和“command”（“ntp.ini”文件所在的路径）。

- hosts: myhosts   
 name: config NTP   
 gather\_facts: False   
   
 tasks:   
   
 # Config NTP function as below:   
 #   
   
 - name: config bmc ntp   
 local\_action: >   
 ibmc category=ConfigNTP command='/etc/ansible/ansible\_ibmc/configFile/ntpCfg/ntp.ini' ibmcip={{ ibmcip }}   
 ibmcuser={{ ibmcuser }} ibmcpswd={{ ibmcpswd }}

执行配置

执行配置命令。

[root@ansible playbooks]# ansible-playbook cfgNTP.yml   
   
PLAY [config NTP] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [config bmc ntp] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com -> localhost]   
ok: [host1.domain.com -> localhost]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0   
host1.domain.com : ok=1 changed=0 unreachable=0 failed=0

查看历史配置记录。

[root@ansible playbooks]# cat /etc/ansible/ansible\_ibmc/report/cfgNTP.log   
[2018-03-22 19:38:20 INFO ] (cfgNTP.py:162)- 192.168.12.10 -- config bmc ntp successfully!   
[2018-03-22 19:38:20 INFO ] (cfgNTP.py:162)- 192.168.3.114 -- config bmc ntp successfully!

## 管理SNMP服务

功能介绍

* 支持查询SNMP服务信息。
* 支持配置trap功能的使能状态、目的IP地址（4组）、trap模式、trap标识和trap严重级别。

### 查询SNMP服务信息（生成json文件）

参数配置

将“/etc/ansible/ansible\_ibmc/playbooks/cfgTrap.yml”文件，中的“command”参数修改为“getinfo”。

- hosts: myhosts   
 name: config snmp trap   
 gather\_facts: no   
   
 tasks:   
   
   
 - name: config snmp   
 local\_action: >   
 ibmc category=CfgSnmpTrap command="getinfo" ibmcip={{ ibmcip }}   
 ibmcuser={{ ibmcuser }} ibmcpswd={{ ibmcpswd }} extraparam={{ community }}

查询SNMP服务信息后自动生成的json文件默认保存在“/etc/ansible/ansible\_ibmc/report”目录中，建议导出json文件后再查看。

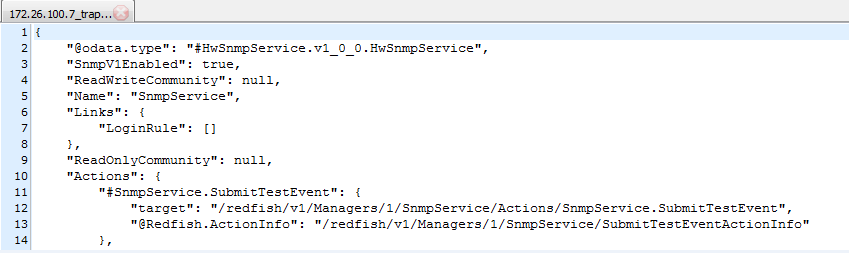
执行配置

执行查询命令。

[root@localhost playbooks]# ansible-playbook cfgTrap.yml   
   
PLAY [config snmp trap] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [config snmp] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com -> localhost]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0

查询SNMP服务信息后自动生成的json文件（如“172.26.100.7\_trapInfo.json”）默认保存在“/etc/ansible/ansible\_ibmc/report”目录中，建议导出json文件后再查看，如图4-4所示。

查询结果



### 配置SNMP Trap

参数配置

1. 需要配置“/etc/ansible/ansible\_ibmc/configFile/snmpCfg/snmpTrap.ini”文件中的如下相关配置：



当“TrapVersion”配置为“V3”时，需要配置“TrapV3User”；当“TrapVersion”配置为“V1”或者“V2C”时需在“/etc/ansible/ansible\_ibmc/playbooks/group\_vars/myhosts”文件中配置“community”。

[snmpTrapNotification]   
# snmp trap config   
# enable or disable snmp trap service,you can choice Y or N   
ServiceEnabled = Y   
# you can choice V1/V2C/V3   
TrapVersion = V3   
TrapV3User = root   
# you can choice OID/EventCode/PreciseAlarm   
TrapMode = OID   
# you can choice BoardSN/ProductAssetTag/HostName   
TrapServerIdentity = HostName   
# you can choice Critical/Major/Minor/Normal   
AlarmServerity = Normal   
TrapDestNum = 4   
   
# Trap destination Server 1   
[trapDest1]   
TrapEnabled = Y   
TrapServerAddress = 192.168.2.196   
TrapServerPort = 161   
   
# Trap destination Server 2   
[trapDest2]   
TrapEnabled = Y   
TrapServerAddress = 192.168.2.6   
TrapServerPort = 1612   
   
# Trap destination Server 3   
[trapDest3]   
TrapEnabled = N   
TrapServerAddress = 192.168.2.7   
TrapServerPort = 1613   
   
# Trap destination Server 4   
[trapDest4]   
TrapEnabled = N   
TrapServerAddress = 192.168.2.8   
TrapServerPort = 1614

1. 修改“/etc/ansible/ansible\_ibmc/playbooks/cfgTrap.yml”文件，包括“hosts”（配置对象）和“command”（“snmpTrap.ini”文件所在的路径）。

- hosts: myhosts   
 name: config snmp trap   
 gather\_facts: no   
   
 tasks:   
   
   
 - name: config snmp   
 local\_action: >   
 ibmc category=CfgSnmpTrap command="/etc/ansible/ansible\_ibmc/configFile/snmpCfg/snmpTrap.ini" ibmcip={{ ibmcip }}   
 ibmcuser={{ ibmcuser }} ibmcpswd={{ ibmcpswd }} extraparam={{ community }}

执行配置

执行配置命令。

[root@foreman playbooks]# ansible-playbook cfgTrap.yml   
   
PLAY [config snmp trap] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [config snmp] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host114.domain.com -> localhost]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host114.domain.com : ok=1 changed=0 unreachable=0 failed=0   
   
[root@foreman playbooks]#

查看历史配置记录。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/report/cfgSnmpReport.log   
[2018-01-25 10:15:49 INFO ] (cfgTrap.py:159)- 192.168.3.114 -- config snmp trap successful!

## 导入导出Profile文件

功能介绍

本功能实现了BIOS和iBMC的Profile配置文件导入和导出功能，支持通过sftp协议将Profile文件从本地导入到服务器，或者将服务器的Profile文件导出到iBMC的“/tmp”目录下。

参数配置

用户需配置“/etc/ansible/ansible\_ibmc/playbooks/serverProfile.yml”文件中的“command”参数。

hosts: myhosts   
 name: import or export server profile   
 gather\_facts: False   
 # default export profile as /tmp/profile.xml if cli has no paramters   
 # export example: 'import;sftp://{{ sftpuser }}:{{ sftppswd }}@192.168.2.156/etc/profile.xml'   
 vars:   
 command: 'export;/tmp/profile.xml'   
 tasks:   
 - name: import or export server profile   
 local\_action: >   
 ibmc category=Profile command={{ command }} ibmcip={{ ibmcip }}   
 ibmcuser={{ ibmcuser }} ibmcpswd={{ ibmcpswd }}

执行操作

将Profile文件导出到iBMC侧的“/tmp”目录下。

[root@foreman playbooks]# ansible-playbook serverProfile.yml   
   
PLAY [import or export server profile] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [export server profile] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host114.domain.com -> localhost]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host114.domain.com : ok=1 changed=0 unreachable=0 failed=0   
   
[root@foreman playbooks]#

查看历史操作记录。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/report/serverProfile.log   
[2018-01-25 10:17:33 INFO ] (serverProfile.py:170)- 192.168.3.114 -- EXPORT:/tmp/profile.xml successful!

## 固件升级

功能介绍

* 支持查询固件信息。
* 支持升级BMC、BIOS和CPLD。

### 查询固件信息（生成json文件）

参数配置

查询固件信息后自动生成的json文件默认保存在“/etc/ansible/ansible\_ibmc/report”目录中，建议导出json文件后再查看。

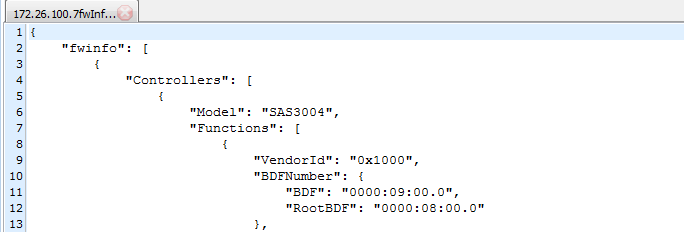
执行操作

执行查询命令。

[root@localhost playbooks]# vi getFwInfo.yml   
[root@localhost playbooks]# ansible-playbook getFwInfo.yml   
   
PLAY [update firm ware] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [update frimware] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host0.domain.com -> localhost]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0

查询固件信息后自动生成的json文件（如“172.26.100.7fwInfo.json”）默认保存在“/etc/ansible/ansible\_ibmc/report”目录中，建议导出json文件后再查看，如图4-5所示。

查询结果



### 升级固件

#### BMC方式

参数配置

修改“/etc/ansible/ansible\_ibmc/playbooks/upgradeFw.yml”文件，包括“hosts”（配置对象）和“command”（待升级固件的镜像文件所在路径）参数。

- hosts: myhosts   
 name: update firm ware   
 gather\_facts: no   
   
 tasks:   
   
 - name: update frimware   
 local\_action: >   
 ibmc category=UpdateFW command='nfs://192.168.2.241/NFSshare/2288V3.hpm' ibmcip={{ ibmcip }}   
 ibmcuser={{ ibmcuser }} ibmcpswd={{ ibmcpswd }}

执行升级

执行升级命令。

[root@localhost playbooks]# ansible-playbook updateFW.yml   
   
PLAY [update firm ware] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [update frimware] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host4.domain.com -> localhost]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host4.domain.com : ok=1 changed=0 unreachable=0 failed=0

查看升级进度。

[root@localhost log]# tail -f /etc/ansible/ansible\_ibmc/log/updateFwLog.log   
[2018-01-25 08:36:41 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:None   
[2018-01-25 08:36:43 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:0%   
[2018-01-25 08:36:45 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:3%   
[2018-01-25 08:36:48 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:4%   
[2018-01-25 08:36:50 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:5%   
[2018-01-25 08:36:53 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:6%   
[2018-01-25 08:36:55 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:9%   
[2018-01-25 08:36:58 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:10%   
[2018-01-25 08:37:01 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:11%   
[2018-01-25 08:37:03 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:13%   
[2018-01-25 08:37:05 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:16%   
[2018-01-25 08:37:08 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:17%   
[2018-01-25 08:37:10 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:18%   
…   
[2018-01-25 08:38:30 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:78%   
[2018-01-25 08:38:33 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:80%   
[2018-01-25 08:38:35 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:82%   
[2018-01-25 08:38:37 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:84%   
[2018-01-25 08:38:40 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:86%   
[2018-01-25 08:38:43 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:91%   
[2018-01-25 08:38:45 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:95%   
[2018-01-25 08:38:47 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:97%   
[2018-01-25 08:38:49 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Running percent:97%   
[2018-01-25 08:38:52 INFO ] (updateFW.py:123)- 192.168.3.114 -- status:Completed percent:100%   
[2018-01-25 08:38:52 INFO ] (updateFW.py:128)- 192.168.3.114 -- update 1288HV5260.hpm successful!

查看历史升级记录。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/report/updateFwReport.log   
[2018-01-25 08:38:52 INFO ] (updateFW.py:129)- 192.168.3.114 -- update 1288HV5260.hpm successful!

#### Smart Provisioning方式

功能介绍

支持升级V5服务器的RAID卡、网卡和硬盘的固件，具体请参考[《FusionServer Tools V200R002C10 Smart Provisioning 用户指南》](http://support.huawei.com/enterprise/zh/doc/DOC1100002507?idPath=7919749%7C9856522%7C9856629%7C21015513)。

参数配置

“/etc/ansible/ansible\_ibmc/configFile/upgradeFwBySpCfg/upgradeFwBySp.ini”文件中“imageurl”（升级包所在路径）参数。将升级包和升级包校验文件上传到同一路径下。若需同时升级多个固件，则需输入各个固件的升级包路径，可以用fw1、fw2来区分不同固件。

[fw1]   
imageurl=nfs://192.168.2.241/NFSshare/inbandfw/BC1SPSCA03-10GE BASE T-X722 FW-V102.zip   
[fw2]   
imageurl=nfs://192.168.2.241/NFSshare/inbandfw/ RAID-SR430C(3108)-FW-V108(4.650.00-6121).zip

执行升级

执行升级命令。

[root@ansible playbooks]# ansible-playbook upgradeFwBySp.yml   
   
PLAY [update firm ware] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [update frimware] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host4.domain.com : ok=1 changed=0 unreachable=0 failed=0

查看升级进度。

[root@localhost log]# tail -f /etc/ansible/ansible\_ibmc/log/upgradFwBySp.log   
[2018-09-05 21:39:18 INFO ] (upgradeFwBySp.py:69)- 192.168.3.10 -- ForceOff system successfully!   
[2018-09-05 21:39:18 INFO ] (upgradeFwBySp.py:78)- 192.168.3.10 -- GetFwUpdateId successfully!   
[2018-09-05 21:39:19 INFO ] (upgradeFwBySp.py:111)- 192.168.3.10 -- spApiSetFwUpgrade successfully!   
[2018-09-05 21:39:40 INFO ] (upgradeFwBySp.py:128)- 192.168.3.10 -- spApiGetFWSource successfully!   
[2018-09-05 21:39:44 INFO ] (upgradeFwBySp.py:149)- 192.168.3.10 -- spAPISetSpService successfully!   
[2018-09-05 21:39:46 INFO ] (upgradeFwBySp.py:164)- 192.168.3.10 -- poweron system successfully!   
…   
[2018-09-05 21:47:22 INFO ] (upgradeFwBySp.py:197)- have got no upgrade result   
[2018-09-05 21:47:33 INFO ] (upgradeFwBySp.py:197)- have got no upgrade result   
…   
[2018-09-05 21:59:39 INFO ] (upgradeFwBySp.py:185)- upgrade has not finished   
[2018-09-05 21:59:47 INFO ] (upgradeFwBySp.py:185)- upgrade has not finished   
[2018-09-05 21:59:56 INFO ] (upgradeFwBySp.py:207)- 192.168.3.10 -- upgrade successfully

查看历史升级记录。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/report/updateFwReport.log   
[2018-09-05 21:59:56 INFO ] (upgradeFwBySp.py:208)- 192.168.3.10 -- upgrade successfully

## RAID配置

功能介绍

支持配置RAID0/RAID10/RAID5/RAID50/RAID6/RAID60，支持查询RAID配置信息，支持配置启动盘，支持创建逻辑盘和配置相关参数配置，同时支持删除RAID配置。

### 查询RAID配置

执行查询命令。

[root@ansible playbooks]# ansible-playbook getRaidInfo.yml   
   
PLAY [get logic device info] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [get logic device info] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host3.domain.com -> localhost]   
ok: [host2.domain.com -> localhost]   
ok: [host0.domain.com -> localhost]   
ok: [host1.domain.com -> localhost]   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0   
host1.domain.com : ok=1 changed=0 unreachable=0 failed=0   
host2.domain.com : ok=1 changed=0 unreachable=0 failed=0   
host3.domain.com : ok=1 changed=0 unreachable=0 failed=0

查看所有服务器的RAID组简要信息。

[root@ansible report]# cat /etc/ansible/ansible\_ibmc/report/getRaidInfoReport.log   
[2018-05-09 22:18:44 INFO ] (getRaidInfo.py:188)- 192.168.3.114 get raid info successfully, totalInfo:   
 ===========================================================   
/redfish/v1/Systems/1/Storages/RAIDStorage0:   
raidmodle:SAS3108   
-unconfigDrivers:   
--HDDPlaneDisk0   
--HDDPlaneDisk1   
--HDDPlaneDisk2   
--HDDPlaneDisk3   
for more info please refer to /etc/ansible/ansible\_ibmc/report/192.168.3.114\_raidInfo.json   
   
[2018-05-09 22:18:49 INFO ] (getRaidInfo.py:188)- 192.168.3.7 get raid info successfully, totalInfo:   
 ===========================================================   
/redfish/v1/Systems/1/Storages/RAIDStorage1:   
raidmodle:SAS3408   
-unconfigDrivers:   
===========================================================   
/redfish/v1/Systems/1/Storages/RAIDStorage0:   
raidmodle:SAS3108   
-LogicalDrive0:   
--raidLevel:RAID0   
--drivers:   
---HDDPlaneDisk3   
---HDDPlaneDisk4   
-unconfigDrivers:   
--HDDPlaneDisk0   
--HDDPlaneDisk1   
--HDDPlaneDisk2   
--HDDPlaneDisk5   
--HDDPlaneDisk6   
--HDDPlaneDisk7   
--HDDPlaneDisk40   
--HDDPlaneDisk41   
for more info please refer to /etc/ansible/ansible\_ibmc/report/192.168.3.7\_raidInfo.json

查看单个服务器的RAID组详细信息。

[root@ansible]# cat /etc/ansible/ansible\_ibmc/report/192.168.3.114\_raidInfo.json   
 "/redfish/v1/Systems/1/Storages/RAIDStorage0": {   
 "@odata.type": "#Storage.v1\_1\_0.Storage",   
 "StorageControllers@odata.count": 1,   
 "Name": "RAIDStorage0",   
 "raidinfo": {   
 "/redfish/v1/Chassis/1/Boards/mainboardRAIDCard1--SAS3108": {   
 "@odata.type": "#HwBoard.v1\_0\_0.HwBoard",   
 "Status": {   
 "State": "Enabled",   
 "Health": "OK"   
 },   
 "PCBVersion": ".B",   
 "CPLDVersion": null,   
 "Name": "mainboardRAIDCard1",   
 "AssetTag": null,   
 "CardNo": 1,   
 "SerialNumber": null,   
 "@odata.id": "/redfish/v1/Chassis/1/Boards/mainboardRAIDCard1",   
 "@odata.context": "/redfish/v1/$metadata#Chassis/Members/1/Boards/Members/$entity",   
 "DeviceLocator": "RAIDCard1",   
 "ManufactureDate": "2016/11/09 Wed 07:35:00",   
 "BoardId": "0x0024",   
 "DeviceType": "RAIDCard",   
 "Location": "mainboard",   
 "BoardName": "BC61ESMLB",   
 "PartNumber": null,   
 "ProductName": "SR530C-M 1G",   
 "Id": "mainboardRAIDCard1",   
 "Manufacturer": "Huawei"   
 }   
 },   
 "@odata.id": "/redfish/v1/Systems/1/Storages/RAIDStorage0",   
 "Actions": {   
 "Oem": {   
 "Huawei": {   
 "#Storage.RestoreStorageControllerDefaultSettings": {   
 "target": "/redfish/v1/Systems/1/Storages/RAIDStorage0/Actions/Oem/Huawei/Storage.RestoreStorageControllerDefaultSettings",   
 "@Redfish.ActionInfo": "/redfish/v1/Systems/1/Storages/RAIDStorage0/RestoreStorageControllerDefaultSettingsActionInfo"   
 }   
 }   
 }   
 },   
 "Drives": [   
 {   
 "@odata.id": "/redfish/v1/Chassis/1/Drives/HDDPlaneDisk0"   
 },   
 {   
 "@odata.id": "/redfish/v1/Chassis/1/Drives/HDDPlaneDisk1"   
 },   
 {   
 "@odata.id": "/redfish/v1/Chassis/1/Drives/HDDPlaneDisk2"   
 },   
 {   
 "@odata.id": "/redfish/v1/Chassis/1/Drives/HDDPlaneDisk3"   
 }

### 删除RAID组

参数配置

修改“/etc/ansible/ansible\_ibmc/playbooks/deleteRaid.yml”文件中的“command”的参数，支持删除一个存储设备的指定RAID组（将“command”设置为“*RAIDStorage0/LogicalDrive0*”，“category”设置为“DelALD”）或所有RAID组（将“command”设置为“*RAIDStorage0*”，“category”设置为“DelLD”），也支持删除所有存储设备的所有RAID组（将“command”设置为“ALL”，“category”设置为“DelLD”）。

---   
- hosts: myhosts   
 name: delete raid   
 gather\_facts: no   
   
   
 tasks:   
   
 - name: delete a raid in a storage   
 local\_action: >   
 ibmc category='DelALD' command='RAIDStorage0/LogicalDrive0' extraparam='' ibmcip={{ ibmcip }}   
 ibmcuser={{ ibmcuser }} ibmcpswd={{ ibmcpswd }}   
   
 #- name: delete ALL raid in a storage   
 # local\_action: >   
 # ibmc category='DelLD' command='RAIDStorage0' extraparam='' ibmcip={{ ibmcip }}   
 # ibmcuser={{ ibmcuser }} ibmcpswd={{ ibmcpswd }}   
 #- name: delete ALL raid in each storage   
 # local\_action: >   
 # ibmc category='DelLD' command='ALL' extraparam='' ibmcip={{ ibmcip }}   
 # ibmcuser={{ ibmcuser }} ibmcpswd={{ ibmcpswd }}

执行操作

执行删除命令。

[root@ansible playbooks]# ansible-playbook deleteRaid.yml   
   
PLAY [delete raid] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [delete a raid] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host1.domain.com -> localhost]   
 to retry, use: --limit @/etc/ansible/ansible\_ibmc/playbooks/deleteRaid.retry   
   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host1.domain.com : ok=1 changed=0 unreachable=0 failed=0

查看历史删除记录。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/report/cfgRaidReport.log   
[2018-05-09 23:52:03 INFO ] (cfgRaid.py:170)- 192.168.3.7 RAIDStorage0/LogicalDrive0 delete LD successful

### 创建RAID组

参数配置

修改“/etc/ansible/ansible\_ibmc/configFile/raidCfg/”目录下的“createRaidtmp.json”文件，可配置单个或多个RAID组，各个参数的配置请参考[《华为服务器 iBMC Redfish 接口说明》](http://support.huawei.com/enterprise/zh/doc/EDOC1000126991)的“创建逻辑盘”章节。



不同型号的RAID卡创建逻辑盘时，RAID卡支持的参数值可能不一致，请参考RAID卡支持的参数进行正确设置。

[root@ansible]# vim createRaidtmp.json   
{   
 "LDlist":[   
 {   
 "RAIDID" :"RAIDStorage0/Volumes" ,   
 "configDic":   
 {   
 "CapacityBytes":104857600000 ,   
 "OptimumIOSizeBytes" : 65536,   
 "Oem":{   
 "Huawei": {   
 "CreateCacheCadeFlag" :false,   
 "Drives" : [0,1],   
 "VolumeName": "testVol1",   
 "VolumeRaidLevel": "RAID0",   
 "DefaultReadPolicy": "ReadAhead",   
 "DefaultWritePolicy" : "WriteThrough",   
 "DefaultCachePolicy" : "CachedIO",   
 "SpanNumber": 1,   
 "AccessPolicy" : "ReadWrite",   
 "DriveCachePolicy" : "Disabled",   
 "InitializationMode" : "UnInit"   
 }   
 }   
 },   
{   
 "RAIDID" :"RAIDStorage0/Volumes" ,   
 "configDic":   
 {   
 "CapacityBytes":104857600000 ,   
 "OptimumIOSizeBytes" : 65536,   
 "Oem":{   
 "Huawei": {   
 "CreateCacheCadeFlag" :false,   
 "Drives" : [2,3],   
 "VolumeName": "testVol2",   
 "VolumeRaidLevel": "RAID1",   
 "DefaultReadPolicy": "ReadAhead",   
 "DefaultWritePolicy" : "WriteThrough",   
 "DefaultCachePolicy" : "CachedIO",   
 "SpanNumber": 1,   
 "AccessPolicy" : "ReadWrite",   
 "DriveCachePolicy" : "Disabled",   
 "InitializationMode" : "UnInit"   
 }   
 }   
 }   
 ]   
}

执行操作

执行创建命令。

[root@localhost playbooks]# ansible-playbook createRaid.yml   
PLAY [config raid] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
TASK [config raid1] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host7.domain.com -> localhost]   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host7.domain.com : ok=1 changed=0 unreachable=0 failed=0

查看历史创建记录。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/report/cfgRaidReport.log   
[2018-05-10 01:52:05 INFO ] (cfgRaid.py:549)- 192.168.3.7 cfgRaid successlly ,result :[{u'RAIDStorage0/Volumes': 'success'}]

### 修改RAID配置

参数配置

修改“/etc/ansible/ansible\_ibmc/configFile/raidCfg/”目录下的“modifyRaid.json”文件，各个参数的配置请参考[《华为服务器 iBMC Redfish 接口说明》](http://support.huawei.com/enterprise/zh/doc/EDOC1000126991)的“修改指定逻辑盘资源属性”章节。

[root@ansible]# vim modifyRaid.json   
{   
 "ldlist":   
 [   
 {   
 "LDID":"RAIDStorage0/Volumes/LogicalDrive0",   
 "LDConfig":   
 {   
 "Oem":   
 {   
 "Huawei":   
 {   
 "VolumeName": "testName1",   
 "DefaultReadPolicy": "NoReadAhead",   
 "DefaultWritePolicy": "WriteThrough",   
 "DefaultCachePolicy": "CachedIO",   
 "BootEnable": true,   
 "BGIEnable": true,   
 "AccessPolicy": "ReadWrite",   
 "SSDCachingEnable": true,   
 "DriveCachePolicy": "Unchanged"   
 }   
 }   
 }   
 },   
 {   
   
 "LDID":"RAIDStorage0/Volumes/LogicalDrive1",   
 "LDConfig":   
 {   
 "Oem":   
 {   
 "Huawei":   
 {   
 "VolumeName": "testName2",   
 "DefaultReadPolicy": "NoReadAhead",   
 "DefaultWritePolicy": "WriteThrough",   
 "DefaultCachePolicy": "CachedIO",   
 "BootEnable": false,   
 "BGIEnable": true,   
 "AccessPolicy": "ReadWrite",   
 "SSDCachingEnable": true,   
 "DriveCachePolicy": "Unchanged"   
 }   
 }   
 }   
   
 }   
 ]   
}

执行操作

执行修改命令。

[root@ansible playbooks]# ansible-playbook modifyRaid.yml   
   
PLAY [modify raid] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
   
TASK [modify raid] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host0.domain.com : ok=1 changed=0 unreachable=0 failed=0

若要查看修改后的RAID配置信息，请先执行**ansible-playbook getRaidInfo.yml**命令获取最新的RAID配置信息，然后查看“/etc/ansible/ansible\_ibmc/report/*192.168.3.7*\_raidInfo.json”文件。

[root@ansible report]# vim 192.168.3.7\_raidInfo.json   
{   
 "/redfish/v1/Systems/1/Storages/RAIDStorage1": {   
 "@odata.type": "#Storage.v1\_1\_0.Storage",   
 "StorageControllers@odata.count": 1,   
 "Name": "RAIDStorage1",   
 "raidinfo": {   
 "/redfish/v1/Chassis/1/Boards/mainboardPCIeCard4(RAID)--SAS3408": {   
 "@odata.type": "#HwBoard.v1\_0\_0.HwBoard",   
 "Status": {   
 "State": "Enabled",   
 "Health": "OK"   
 },   
 "PCBVersion": ".A",   
 "CPLDVersion": null,   
 "Name": "mainboardPCIeCard4(RAID)",   
 "AssetTag": null,   
 "CardNo": 4,   
 "SerialNumber": null,   
 "@odata.id": "/redfish/v1/Chassis/1/Boards/mainboardPCIeCard4(RAID)",   
 "@odata.context": "/redfish/v1/$metadata#Chassis/Members/1/Boards/Members/$entity",   
 "DeviceLocator": "PCIeCard4(RAID)",   
 "ManufactureDate": "2018/03/18 Sun 16:41:00",   
 "BoardId": "0x0059",   
 "DeviceType": "PCIeCard",   
 "Location": "mainboard",   
 "BoardName": null,   
 "PartNumber": null,   
 "ProductName": null,   
 "Id": "mainboardPCIeCard4(RAID)",   
 "Manufacturer": "LSI Logic / Symbios Logic"   
 }   
 },   
 "@odata.id": "/redfish/v1/Systems/1/Storages/RAIDStorage1",   
 "Actions": {   
 "Oem": {   
 "Huawei": {   
 "#Storage.RestoreStorageControllerDefaultSettings": {   
 "target": "/redfish/v1/Systems/1/Storages/RAIDStorage1/Actions/Oem/Huawei/Storage.RestoreStorageControllerDefaultSettings",   
 "@Redfish.ActionInfo": "/redfish/v1/Systems/1/Storages/RAIDStorage1/RestoreStorageControllerDefaultSettingsActionInfo"   
 }   
 }   
 }   
 },   
 "Drives": [],   
 "driverinfo": {},   
 "@odata.context": "/redfish/v1/$metadata#Systems/Members/1/Storages/Members/$entity",   
 "Volumes": {   
 "@odata.id": "/redfish/v1/Systems/1/Storages/RAIDStorage1/Volumes"   
 },   
 "StorageControllers": [   
 {   
 "Status": {   
 "State": "Enabled",   
 "Health": "OK"   
 },   
 "Description": "RAID Controller",   
"192.168.3.7\_raidInfo.json" [noeol] 876L, 39642C

查看历史修改记录。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/report/cfgRaidReport.log   
[2018-05-11 01:04:55 INFO ] (cfgRaid.py:435)- 192.168.3.7 modify raid successlly ,result :[{u'RAIDStorage0/Volumes/LogicalDrive0': 'success'}, {u'RAIDStorage0/Volumes/LogicalDrive1': 'success'}]

## OS部署

功能介绍

支持部署的OS如下：

* Windows：Windows 2012 R2、Windows 2016。
* Linux：CenteOS系列、RedHat系列、SUSE系列。
* VMware：VMware 5.5、VMware 6.0、VMware 6.5。



待部署OS的服务器需提前完成RAID配置。

### ServiceCD2.0方式

参数配置

“/etc/ansible/ansible\_ibmc/configFile/deployCfg/ServiceCD/deployConfig.ini”文件用于配置OS的类型、OS镜像文件路径、serviceCD2.0镜像文件路径，其中OS类型需按照下面的格式进行填写（所支持的OS类型已全部列出）。

;WINDOWS   
;Win2008\_R2\_x64   
;Win2012\_x64   
;Win2012\_R2\_x64   
;Win2016\_x64   
;UBUNTU   
;Ubuntu14.04\_x64   
;Ubuntu16.04\_x64   
   
[config]   
osType = Win2012\_R2\_x64   
osImg = nfs://192.168.2.241/NFSshare/Windows2012R2.iso   
serviceImg = nfs://192.168.2.241/NFSshare/ServiceCD133.iso

“/etc/ansible/ansible\_ibmc/configFile/deployCfg/ServiceCD/”目录下的OS部署的模板需参考表4-1进行配置。



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

OS部署模板

| **文件** | **支持的OS类型** | **支持配置的参数** |
| --- | --- | --- |
| CentOS.xml | CentOS系列 | 初始密码、hostname、owername、orgname、部署系统位置、分区、语言、时区、是否安装iBMA、RPM包 |
| RedHat.xml | Red Hat系列 | 初始密码、hostname、owername、orgname、部署系统位置、分区、语言、时区、是否安装iBMA、RPM |
| SLES11SP1\_64.xml | SLES 11 SP 1 | 初始密码、hostname、owername、orgname、部署系统位置、分区、语言、时区、是否安装iBMA、RPM |
| SLES11SP3\_64.xml | SLES 11 SP 3 | 初始密码、hostname、owername、orgname、部署系统位置、分区、语言、时区、是否安装iBMA、RPM |
| VM5.0.xml | VMware 5.0 | 初始密码、hostname、owername、orgname、部署系统位置、分区、语言、时区、是否安装iBMA |
| VM5.5.xml | VMware 5.5 | 初始密码、hostname、owername、orgname、部署系统位置、分区、语言、时区、是否安装iBMA |
| VM6.0.xml | VMware 6.0 | 初始密码、hostname、owername、orgname、部署系统位置、分区、语言、时区、是否安装iBMA |
| VM6.5.xml | VMware 6.5 | 初始密码、hostname、owername、orgname、部署系统位置、分区、语言、时区、是否安装iBMA |
| win2008r2.xml | Windows 2008 R2 | 初始密码、hostname、owername、orgname、部署系统位置、分区、语言、时区、是否安装iBMA |
| win2012r2.xml | Windows 2012 R2 | 初始密码、hostname、owername、orgname、部署系统位置、分区、语言、时区、是否安装iBMA |
| win2012.xml | Windows 2012 | 初始密码、hostname、owername、orgname、部署系统位置、分区、语言、时区、是否安装iBMA |
| win2016.xml | Windows 2016 | 初始密码、hostname、owername、orgname、部署系统位置、分区、语言、时区、是否安装iBMA |

修改“/etc/ansible/ansible\_ibmc/playbooks/deployOsByServiceCD.yml”文件，包括“hosts”（配置对象）和“command”（“deployConfig.ini”文件所在路径）参数，需要注意的是目前只支持cifs和nfs两种协议来读取文件。

---   
   
- hosts: myhosts   
 name: deploy os   
 gather\_facts: no   
 tasks:   
 - name: deploy os   
 local\_action: >   
 ibmc category=DeployOS command='/etc/ansible/ansible\_ibmc/configFile/deployCfg/ServiceCD/deployConfig.ini' ibmcip={{ ibmcip }}   
 ibmcuser={{ ibmcuser }} ibmcpswd={{ ibmcpswd }}

执行部署

执行部署命令。

[root@localhost playbooks]# ansible-playbook deployOsByServiceCD.yml   
PLAY [deploy os] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
TASK [deploy os] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
ok: [host7.domain.com -> localhost]   
PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
host7.domain.com : ok=1 changed=0 unreachable=0 failed=0

查看历史部署过程日志。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/log/deployOsLog.log   
[2018-05-18 22:11:34 INFO ] (deployOs.py:344)- 192.168.4.7 -- ServiceCD133.iso;rhel73.iso;/etc/ansible/ansible\_ibmc/configFile/deployCfg/ServiceCD/RedHat.xml   
[2018-05-18 22:11:37 INFO ] (deployOs.py:366)- 192.168.4.7 -- read for bmc info:operator:eSight;osType:RHEL7U3\_x64   
[2018-05-18 22:11:37 INFO ] (deployOs.py:370)- 192.168.4.7 -- is connect:False   
[2018-05-18 22:11:37 INFO ] (deployOs.py:379)- 192.168.4.7 -- set boot device to CD!   
[2018-05-18 22:11:39 INFO ] (deployOs.py:386)- 192.168.4.7 -- bootdevice is:Cd   
[2018-05-18 22:11:39 INFO ] (deployOs.py:237)- 192.168.4.7 -- mount ServiceCD133.iso successful   
[2018-05-18 22:11:49 INFO ] (deployOs.py:400)- 192.168.4.7 -- reboot system successfully!   
[2018-05-18 22:12:09 INFO ] (deployOs.py:415)- 192.168.4.7 -- loop:1 ret: operator:eSight;osType:RHEL7U3\_x64   
[2018-05-18 22:12:30 INFO ] (deployOs.py:415)- 192.168.4.7 -- loop:2 ret: operator:eSight;osType:RHEL7U3\_x64

查看历史部署结果日志。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/report/deployOsReport.log   
[2018-01-25 07:06:36 INFO ] (deployOs.py:786)- 192.168.3.114 -- install OS(Win2012\_R2\_x64) successfully!

### Smart Provisioning方式

此方式只支持V5服务器，支持部署的操作系统请参考[《FusionServer Tools V200R002C10 Smart Provisioning 用户指南》](http://support.huawei.com/enterprise/zh/doc/DOC1100002507?idPath=7919749%7C9856522%7C9856629%7C21015513)。

参数配置

* “/etc/ansible/ansible\_ibmc/configFile/deployCfg/SP/deployConfig.ini”文件用于配置OS镜像文件路径、“*OS*.json”文件路径。

; This is the config file of OS deployment for Huawei Smart Provision Tools.   
   
[ConfigOS]   
# Input the os image file path on NFS server   
osImg = nfs://192.168.2.241/NFSshare/Windows2012R2.iso   
   
# Input the os config json file path on "/etc/ansible/ansible\_ibmc/configFile/SP"   
osConfig = /etc/ansible/ansible\_ibmc/configFile/deployCfg/SP/Win2012\_R2.json

“/etc/ansible/ansible\_ibmc/configFile/deployCfg/SP/”目录下OS部署的json模板需参考表4-2进行配置，所支持的OS类型已全部列出。



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

OS部署模板

| **文件** | **支持的OS类型** | **支持配置的参数** |
| --- | --- | --- |
| CentOS7U3.json | CentOS系列 | 时区、键盘语言、是否检查固件、手动分区、自动分区、自动指定安装位置、安装自带软件、网络配置（设备：匹配网卡名称或Mac地址；IPv4地址设定；IPv6地址设定；DNS服务器地址设定） |
| RHEL7U3.json | Red Hat系列 | 时区、键盘语言、是否检查固件、手动分区、自动分区、自动指定安装位置、安装自带软件、网络配置（设备：匹配网卡名称或Mac地址；IPv4地址设定；IPv6地址设定；DNS服务器地址设定） |
| SLES12SP3.json | SLES系列 | 时区、键盘语言、是否检查固件、手动分区、自动分区、自动指定安装位置、安装自带软件、网络配置（设备：匹配网卡名称或Mac地址；IPv4地址设定；IPv6地址设定；DNS服务器地址设定） |
| ESXi6.5.json | VMware ESXi系列 | 时区、键盘语言、是否检查固件、手动分区、自动分区、自动指定安装位置、安装自带软件、网络配置（设备：匹配网卡名称或Mac地址；IPv4地址设定；IPv6地址设定；DNS服务器地址设定） |
| Win2012\_R2.json | Windows系列 | 时区、键盘语言、是否检查固件、手动分区、自动分区、自动指定安装位置、安装自带软件、网络配置（设备：匹配网卡名称或Mac地址；IPv4地址设定；IPv6地址设定；DNS服务器地址设定） |

* 修改“/etc/ansible/ansible\_ibmc/playbooks/deployOsBySp.yml”文件，包括“hosts”（配置对象）和“command”（“deployConfig.ini”文件所在路径）参数，需要注意的是目前只支持cifs和nfs两种协议来读取文件。

---   
   
- hosts: myhosts   
 name: deploy os   
 gather\_facts: no   
 tasks:   
 - name: deploy os   
 local\_action: >   
 ibmc category=DeployOSBySp command='/etc/ansible/ansible\_ibmc/configFile/deployCfg/SP/deployConfig.ini' ibmcip={{ ibmcip }}   
 ibmcuser={{ ibmcuser }} ibmcpswd={{ ibmcpswd }}

执行部署

执行部署命令。

[root@localhost playbooks]# ansible-playbook deployOsBySp.yml   
 PLAY [deploy os] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
 TASK [deploy os] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
 ok: [host7.domain.com -> localhost]   
 PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
 host7.domain.com : ok=1 changed=0 unreachable=0 failed=0

查看历史部署过程日志。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/log/deploySPOsLog.log   
[2018-05-18 21:50:10 INFO ] (deployOsBySp.py:57)- 192.168.3.114 -- the iBMC version is greater than 3.00, match the redfish interface requirement, continue!   
   
[2018-05-18 21:50:10 INFO ] (deployOsBySp.py:98)- 192.168.3.114 -- the SP version is greater than 1.09, match the redfish interface requirement!   
   
[2018-05-18 21:50:10 INFO ] (deployOsBySp.py:57)- 192.168.3.7 -- the iBMC version is greater than 3.00, match the redfish interface requirement, continue!   
   
[2018-05-18 21:50:11 INFO ] (deployOsBySp.py:423)- 192.168.3.114 --vmm is connected:False   
[2018-05-18 21:50:11 INFO ] (deployOsBySp.py:98)- 192.168.3.7 -- the SP version is greater than 1.09, match the redfish interface requirement!   
   
   
==> pmLog.log <==   
[2018-05-18 21:50:11 INFO ] (powerManage.py:82)- 192.168.3.114 -- set system power off successful!   
   
   
==> deploySPOsLog.log <==   
[2018-05-18 21:50:11 INFO ] (deployOsBySp.py:433)- 192.168.3.114 -- Power off Operation System successfully!   
[2018-05-18 21:50:11 INFO ] (deployOsBySp.py:423)- 192.168.3.7 --vmm is connected:False   
   
==> pmLog.log <==   
[2018-05-18 21:50:12 INFO ] (powerManage.py:82)- 192.168.3.7 -- set system power off successful!   
   
   
==> deploySPOsLog.log <==   
[2018-05-18 21:50:12 INFO ] (deployOsBySp.py:433)- 192.168.3.7 -- Power off Operation System successfully!   
[2018-05-18 21:50:22 INFO ] (deployOsBySp.py:275)- 192.168.3.114 -- post os config parament successfully!   
   
[2018-05-18 21:50:22 INFO ] (deployOsBySp.py:275)- 192.168.3.7 -- post os config parament successfully!   
   
[2018-05-18 21:50:23 INFO ] (deployOsBySp.py:314)- 192.168.3.114 -- set SP result finished successful!   
   
[2018-05-18 21:50:23 INFO ] (deployOsBySp.py:314)- 192.168.3.7 -- set SP result finished successful!   
   
[2018-05-18 21:50:24 INFO ] (deployOsBySp.py:353)- 192.168.3.114 -- set boot device as SP successful!   
   
[2018-05-18 21:50:24 INFO ] (deployOsBySp.py:353)- 192.168.3.7 -- set boot device as SP successful!   
   
[2018-05-18 21:50:24 INFO ] (deployOsBySp.py:192)- 192.168.3.114 -- mount rhel73.iso successful   
[2018-05-18 21:50:25 INFO ] (deployOsBySp.py:192)- 192.168.3.7 -- mount rhel73.iso successful   
   
==> pmLog.log <==   
[2018-05-18 21:50:34 INFO ] (powerManage.py:60)- 192.168.3.114 -- set system power on successful!   
   
   
==> deploySPOsLog.log <==   
[2018-05-18 21:50:34 INFO ] (deployOsBySp.py:481)- 192.168.3.114 -- reboot system successfully!   
   
==> pmLog.log <==   
[2018-05-18 21:50:36 INFO ] (powerManage.py:60)- 192.168.3.7 -- set system power on successful!   
   
   
==> deploySPOsLog.log <==   
[2018-05-18 21:50:36 INFO ] (deployOsBySp.py:481)- 192.168.3.7 -- reboot system successfully!   
[2018-05-18 21:50:40 INFO ] (deployOsBySp.py:385)- 192.168.3.114: SP Status is Init   
   
[2018-05-18 21:50:40 INFO ] (deployOsBySp.py:501)- 192.168.3.114 -- loopInstall:1 SPStatus:Init, OSProgress:, OSStatus:, OSStep:, OSErrorInfo:   
   
[2018-05-18 21:50:40 INFO ] (deployOsBySp.py:503)- 192.168.3.114 -- SP is initial, please wait!   
[2018-05-18 21:50:45 INFO ] (deployOsBySp.py:385)- 192.168.3.7: SP Status is Init

查看历史部署结果日志。

[root@localhost playbooks]# cat /etc/ansible/ansible\_ibmc/report/deploySPOsReport.log   
[2018-05-11 18:16:04 INFO ] (deployOsBySp.py:274)- 192.168.3.7 -- post os config parament successfully!   
[2018-05-11 18:16:05 INFO ] (deployOsBySp.py:274)- 192.168.3.114 -- post os config parament successfully!   
[2018-05-11 18:53:08 INFO ] (deployOsBySp.py:509)- 192.168.3.114 -- os install successfully   
[2018-05-11 19:01:03 INFO ] (deployOsBySp.py:509)- 192.168.3.7 -- os install successfully

# FAQ

[5.1 如何配置系统的多个用户均可以使用Ansible](#_ZH-CN_TOPIC_0195360757)

## 如何配置系统的多个用户均可以使用Ansible

新建Ansible用户和组。

**useradd Ansible**

修改Ansible目录所有者。

**cd /etc/ansible**

**Chown -R Ansible:Ansible ansible\_ibmc**

修改Ansible目录权限。

**cd /etc/ansible**

**chmod -R 775 ansible\_ibmc**

添加其他用户至Ansible组（如“testansible”）。

**useradd -g Ansible testansible**

----结束