**openUBMC**

# 管理系统技术白皮书

**发布日期　2026-06-22**

# 前言

## 概述

本文档详细地描述了服务器openUBMC管理系统的主要特性，让用户对openUBMC有一个深入细致的了解。

## 读者对象

本文档主要适用于以下人员：

- 社区开发者。
- 整机厂商系统工程师。
- 部件厂商系统工程师

## 修改记录

| 文档版本 | 发布日期 | 修改说明 |
| --- | --- | --- |
| 01 | 2025-03-20 | 第一次正式发布。 |
| 02 | 2025-06-30 | 1. 新增可观测服务说明<br>2. 新增并行升级功能说明<br>3. 修复文档描述不准确信息 |
| 03 | 2025-09-30 | 1. 补充基础安全相关的LDAP认证、双因素认证等功能 |
| 04 | 2025-12-30 | 1. 虚拟KVM中新增VNC功能说明<br>2. 更新可观测服务说明<br>3. 更新安全管理说明，调整账户与身份认证目录结构<br>4. 新增目录服务功能说明<br>5. 新增故障管理章节，新增光模块和硬盘故障管理能力说明<br>6. 修复文档描述不准确信息 |

| 文档版本 | 发布日期 | 修改说明 |
| --- | --- | --- |
| 05 | 2026-03-20 | 1. 新增RAID卡和硬盘固件升级 |
| 06 | 2026-06-22 | 1. 新增线缆故障管理特性说明<br>2. 新增CPLD无感升级说明<br>3. 新增网卡和NVMe盘带外升级能力<br>4. 新增硬盘故障检测及自愈<br>5. 新增固件恢复/固件整包升级/固件暂存升级章节<br>6. 修改配置导入导出章节<br>7. 修改密码配置策略章节<br>8. 修复部分章节图片大小问题 |

# 目 录

# 1 产品简介



## 1.1 概述

openUBMC兼容服务器业界管理标准IPMI 、SNMP、Redfish，具备键盘、鼠标和视频的重定向、文本控制台的重定向、远程虚拟媒体、高可靠的硬件监控和管理等功能。openUBMC提供了丰富的特性支持。其主要特性有：

- 丰富的管理接口
  提供IPMI/CLI/SNMP/Redfish/WEB管理接口，满足多种方式的系统集成需求。
- 兼容DCMI1.5/IPMI1.5/ IPMI2.0
  提供标准的管理接口，可被标准管理系统集成。
- 故障监控和诊断
  基于部件的精准故障诊断，方便部件故障定位和更换。
- 虚拟KVM和虚拟媒体
  提供方便的远程维护手段。
- 基于Web界面的用户接口
  可以通过简单的界面操作快速完成设置和查询任务。
- 系统崩溃时临终截屏与录像
  分析系统崩溃原因不再无处下手。
- 屏幕快照和屏幕录像
  让定时巡检、操作过程记录及审计变得简单轻松。
- 支持DNS/LDAP/LLDP
  域管理、目录服务、管理网口链路层发现协议报文发送，简化服务器管理网络。
- 软件双镜像备份
  提高系统的安全性，即使当前运行的软件完全崩溃，也可以从备份镜像启动。
- RAID带外管理
  支持RAID的带外监控和配置并支持1880 RAID带外升级能力，提升了RAID配置效率和管理能力。

- 安全管理

  从接入、账号、传输、存储四个维度保障服务器管理的安全，让您用得放心。
- 网卡带外管理

  支持网卡带外升级框架，提升了网卡的升级效率，解决依赖带内通道的痛点。
- 硬盘带外管理

  支持1880 RAID卡管理的SAS/SATA盘的带外升级，实现了升级盘固件过程中不会中断业务，并解决依赖带内通道的痛点。

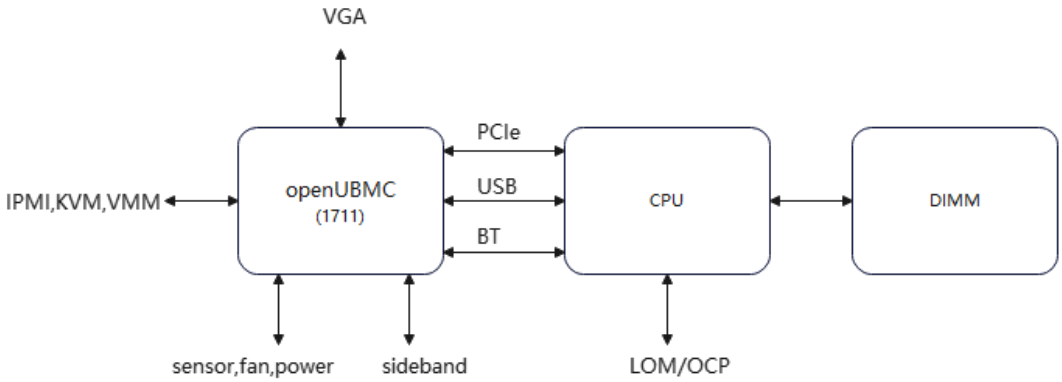
# 1.2 系统架构

如图1-1所示，openUBMC硬件芯片当前采用Hi1711芯片，Hi1711是一款针对服务器硬件管理的板级管理 BMC 芯片，包括一个最高主频为1.0 GHz 的四核 A55 CPU，一个协处理 M3（主频 200MHz ）及安全核 M3 （主频200MHz，支持安全启动）。芯片支持eFuse、支持远程 KVM，支持 IPMI 管理接口，支持 PCIE 收发 MCTP 报文，支持本地显示 VGA，GE 网口、RMII 接口，以及其它丰富的板级管理和外设接口。

具体如下：

- openUBMC的KVM模块通过VGA接口接收来自业务系统的视频信息，经过压缩后再通过网络将压缩数据传输到远程KVM客户端进行解压还原。此外KVM模块接收远程KVM客户端的键盘鼠标数据，通过模拟的USB键盘鼠标设备将数据传输到业务系统，实现远程的键盘鼠标控制。
- openUBMC的VMM模块将光驱等本地资源虚拟为服务器的USB设备。
- openUBMC提供BT系统接口与业务系统通信，支持标准的IPMI管理。
- openUBMC对外提供GE以太网络接口，支持通过网络使用IPMI，HTTPS等协议进行远程管理操作。
- openUBMC通过传感器和软件告警监控实现了对服务器的温度、电压状态全面监控，并且提供对服务器的风扇和电源的智能管理。
- openUBMC支持最新的边带网络技术（Side band，如：NCSI）以及VLAN网络功能，通过边带网络可以支持更加灵活的管理组网。

**图 1-1** openUBMC 系统架构



# 2 支持功能

openUBMC以其丰富的特性，可以有效提升管理效率，降低运营成本。

- openUBMC是独立开发的具有完全自主知识产权的高级服务器远程管理软件。它支持键盘、鼠标和视频的重定向、文本控制台的重定向、远程虚拟媒体（可将终端的光驱、文件夹映射到服务器）和基于IPMI/Redfish的硬件监控和管理功能。可靠性能力强，支持双镜像备份的软件。

  openUBMC提供了丰富的用户接口，如命令行、基于Web界面的用户接口、IPMI集成接口、SNMP集成接口、Redfish集成接口，并且所有用户接口都采用了认证机制和高度安全的加密算法，保证接入和传输的安全性。
- openUBMC对服务器进行了全面精细的监控，并且提供了丰富的告警和详细的日志。能够独立显示主板电源故障、CPU的内核温度、电压、硬盘故障、风扇转速及温度故障、系统电源故障、总线故障、系统宕机故障等。同时还提供了CPU、内存、网卡和硬盘等各类部件信息的查询。同时支持对告警日志、错误日志、部件信息等实现一键收集辅助问题定位。
- openUBMC能够在服务器宕机的时候自动保存宕机之前屏幕上输出的最后的信息，用于故障的定位。还支持即时的屏幕快照，第三方程序可以设置定时或周期性地进行屏幕截屏，不需要手工定时去查看服务器，为维护人员节省大量时间。

# 2.1 丰富的管理接口

openUBMC是一个遵循行业管理规范的带外单机管理系统，是数据中心管理网络中的一个子节点，承载着管理、控制和诊断服务器的任务，需要对外提供各种人机接口和机机接口，以满足各种服务器管理场景的应用和集成需求。

openUBMC的框架分四层，即：接口层、资源协作接口、组件集、框架层，接口层主要提供各种接口，包括用户接口（Web和CLI）和机机接口（SNMP、IPMI和Redfish）；资源协作接口是从系统视角进行的系统资源和资源关系的抽象定义；组件集是实现系统资源接口、提供服务的集合；框架层包含：协议库、硬件访问代理、CSR（Component Self-Description Record）解析器、linux内核和驱动。

**图 2-1** openUBMC 管理架构图

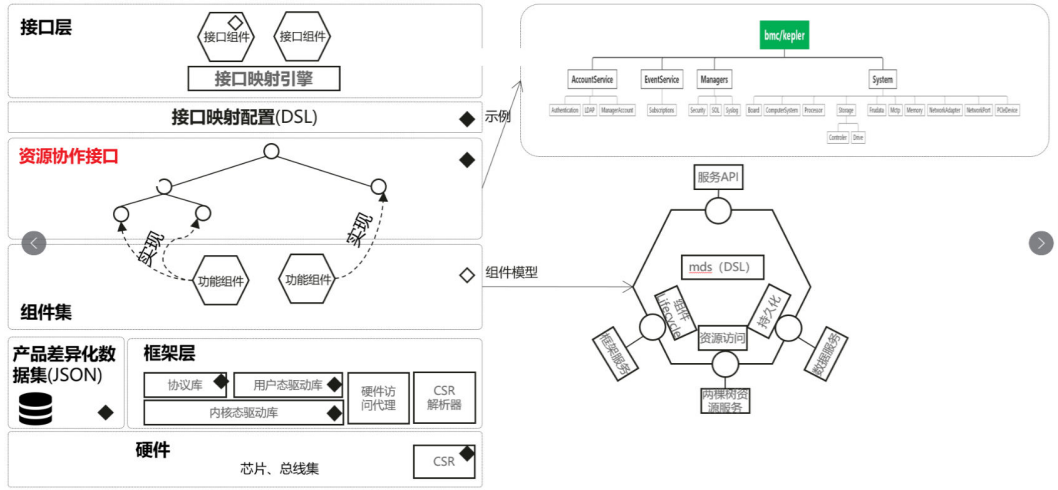



**表 2-1** 主要集成接口对比

| 接口 | 难度 | 集成工作量 | 兼容性 | 安全性 | 性能 | 架构先进性 | 应用 |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Redfish | 易，使用最流行python解析型语言编程，json格式输入输出 | 较小，json输入输出，不用额外解析 | 好，规范定义较全，意在替代IPMI | 高，基于HTTPS协议，支持各种安全的加密、完整性、鉴权算法 | 好，一次交互可以获取整个资源 | 好，所有事物都抽象为资源，有唯一URI，面向对象架构 | 业界互联网/网管都选择或准备选择REST+Python，应用广泛 |

| 接口 | 难度 | 集成工作量 | 兼容性 | 安全性 | 性能 | 架构先进性 | 应用 |
| --- | --- | --- | --- | --- | --- | --- | --- |
| SNMP | 中，需要理解MIB库和OID、SNMP规范 | 中，依赖MIB库进行解析 | 较差，规范定义较少，基本都是网络相关标准节点 | 低，支持的安全算法有限，目前支持鉴权算法MD5/SHA/SHA256/SHA384/SHA512，加密算法仅支持DES/AES/AES256，不支持域账号访问 | 较差，每次交互只能获取一个信息，最大限制为4K字节 | 较差，面向节点，缺乏层次和关联 | 在网络交换设备管理领域比较流行，总体占有率一般 |
| IPMI | 难，C语言编程，掌握难度大 | 大，二进制输出，不友好且解析工作量大 | 较差，规范定义较少且很久未更新 | 低，支持的安全算法有限，出现过安全漏洞，不支持域账号访问 | 较差，每次交互只能获取一个信息，带内通道最大限制为255字节 | 较差，面向命令，缺乏层次和关联 | 仅服务器行业比较流行，总体占有率不高 |

**说明**

基于上述优劣对比，后续服务器管理软件对外集成接口以Redfish接口为主，主动规划并及时跟进DMTF的Redfish规范更新。

## 2.1.1 IPMI 管理接口

openUBMC兼容IPMI 1.5/IPMI 2.0规范，使用第三方工具（如：ipmitool），通过基于BT接口的BT协议或LAN通道的UDP/IP协议实现对服务器的有效管理。基于BT接口时，ipmitool等工具必须运行在服务器本机的操作系统上；而基于LAN时，ipmitool等工具可以远程管理服务器，openUBMC支持AES-CBC-128加密算法，以及HMAC-SHA1/HMAC-SHA256鉴权和完整性校验算法。支持Windows和Linux系统下第三方工具。

IPMI规范所有服务器厂商都支持，并且由于支持本机内部通道通讯，本机内部通道通讯时支持免鉴权，在服务器管理行业应用较广泛，特别是早期的带内管理场景。

对于管理网络和业务网络具有隔离诉求的场景，openUBMC支持配置黑名单和白名单机制屏蔽带内BT通道的IPMI命令下发。白名单和黑名单的配置范围包括通道号、网络功能码、命令字、子命令、参数等选项，支持添加、删除和查询。功能禁用时带内和带外正常通信，启动白名单时仅允许白名单中配置的命令在带内通道下发；启动黑名单时禁止黑名单中配置的命令在带内通道下发。默认启用为黑名单模式，支持切换为白名单模式。

以下以ipmitool工具举例说明：

- ipmitool命令格式：**ipmitool [interface] [parameter] <command>**
- ipmitool命令可设置的接口包括：

```
Interfaces:
open              Linux OpenIPMI Interface [default]
imb               Intel IMB Interface
lan               IPMI v1.5 LAN Interface
lanplus           IPMI v2.0 RMCP+ LAN Interface
```

- ipmitool命令可设置的参数包括：

```
Parameters:
-h               This help
-V               Show version information
-v               Verbose (can use multiple times)
-c               Display output in comma separated format
-d  N             Specify a /dev/ipmiN device to use (default=0)
-I  intf          Interface to use
-H  hostname       Remote host name for LAN interface
-p  port          Remote RMCP port [default=623]
-U  username       Remote session username
-f  file          Read remote session password from file
-S  sdr           Use local file for remote SDR cache
-a               Prompt for remote password
-e  char          Set SOL escape character
-C  ciphersuite    Cipher suite to be used by lanplus interface
-k  key           Use Kg key for IPMIv2 authentication
-y  hex_key       Use hexadecimal-encoded Kg key for IPMIv2 authentication
-L  level         Remote session privilege level [default=ADMINISTRATOR] Append a '+' to use name/
privilege lookup in RAKP1
-A  authtype       Force use of auth type NONE, PASSWORD, MD2, MD5 or OEM
-P  password       Remote session password
-E               Read password from IPMI_PASSWORD environment variable
-K               Read kgkey from IPMI_KGKEY environment variable
-m  address        Set local IPMB address
-b  channel        Set destination channel for bridged request
-t  address        Bridge request to remote target address
-B  channel        Set transit channel for bridged request (dual bridge)
-T  address        Set transit address for bridge request (dual bridge)
-l  lun           Set destination lun for raw commands
-o  oemtype        Setup for OEM (use 'list' to see available OEM types)
-O  seloem         Use file for OEM SEL event descriptions
```

- ipmitool可执行的操作包括：

```
Commands:
raw              Send a RAW IPMI request and print response
i2c              Send an I2C Master Write-Read command and print response
spd              Print SPD info from remote I2C device
lan              Configure LAN Channels
chassis           Get chassis status and set power state
power             Shortcut to chassis power commands
event            Send pre-defined events to MC
mc               Management Controller status and global enables
sdr              Print Sensor Data Repository entries and readings
sensor            Print detailed sensor information
fru              Print built-in FRU and scan SDR for FRU locators
gendev            Read/Write Device associated with Generic Device locators sdr
```

```
sel             Print System Event Log (SEL)
pef             Configure Platform Event Filtering (PEF)
sol             Configure and connect IPMIv2.0 Serial-over-LAN
tsol            Configure and connect with Tyan IPMIv1.5 Serial-over-LAN
isol            Configure IPMIv1.5 Serial-over-LAN
user            Configure Management Controller users
channel         Configure Management Controller channels
session         Print session information
sunoem          OEM Commands for Sun servers
kontronoem      OEM Commands for Kontron devices
picmg           Run a PICMG/ATCA extended cmd
fwum            Update IPMC using Kontron OEM Firmware Update Manager
firewall        Configure Firmware Firewall
delloem         OEM Commands for Dell systems
shell           Launch interactive IPMI shell
exec            Run list of commands from file
set             Set runtime variable for shell and exec
hpm             Update HPM components using PICMG HPM.1 file
ekanalyzer      Run FRU-Ekeying analyzer using FRU files
```

- ipmitool命令举例：查询openUBMC上所有本地用户

  基于BT

  **ipmitool user list**

  基于LAN

  **ipmitool -H *.*.*.* -I lanplus -C** <算法套件号> **-U** <用户名> **-P** <密码> **user list 1**

  **说明**

  - H：openUBMC 网口IP地址
  - I：传输协议，lan：不加密，lanplus：加密
  - C：传输协议指定为lanplus时使用的算法套件
  - U：openUBMC本地用户名
  - P：openUBMC本地用户密码

## 2.1.2 SNMP 管理接口

简单网络管理协议（以下简称SNMP）是管理进程（NMS）和代理进程（Agent）之间的通信协议。它规定了在网络环境中对设备进行监视和管理的标准化管理框架、通信的公共语言、相应的安全和访问控制机制。

openUBMC提供了SNMP的编程接口，支持SNMP Get/Set/Trap操作。通过第三方管理软件调用SNMP接口可以方便地对服务器集成管理。SNMP代理支持V1/V2C/V3版本，出厂默认只启用V3版本。SNMP V1/V2C的Get/Set操作可以使用不同的团体名；SNMP V3的鉴权算法支持选择MD5或SHA、SHA256、SHA384、SHA512，加密算法支持选择DES或AES、AES256，安全用户名与登录用户名相同。SNMP V3安全用户与其他接口（Web、CLI、IPMI LAN）共用一套本地用户。

SNMP接口应用场景：

- 场景1——基于开源工具的管理

  直接使用第三方的MIB图形工具（如MG-SOFT MIB Browser）和命令行工具基于SNMP协议对每个MIB节点进行操作，通常用于测试或临时的服务器远程管理和维护。

- 场景2——简单集成管理

  网管软件将SNMP MIB定义文档编译后导入，即可通过SNMP接口管理服务器，并可对重要的信息配置触发脚本以及对Trap事件进行重新映射；目前已和业界常用的CA、IBM System Director、HP SIM网管软件进行了对接验证。

- 场景3——深度集成管理

  网管支持插件方式，针对不同服务器厂商开发不同的集成管理插件，插件接收网管的操作命令并通过SNMP接口与openUBMC交互进行查询和设置信息，然后按照网管与插件接口格式返回给网管进行展示；目前已为业界常用的Vmware Vcenter、微软System Center网管软件开发了插件。

## 2.1.3 Redfish 管理接口

REST（Representational State Transfer）是一种针对网络应用的设计和开发方式，可以降低开发的复杂性，提高系统的可伸缩性。

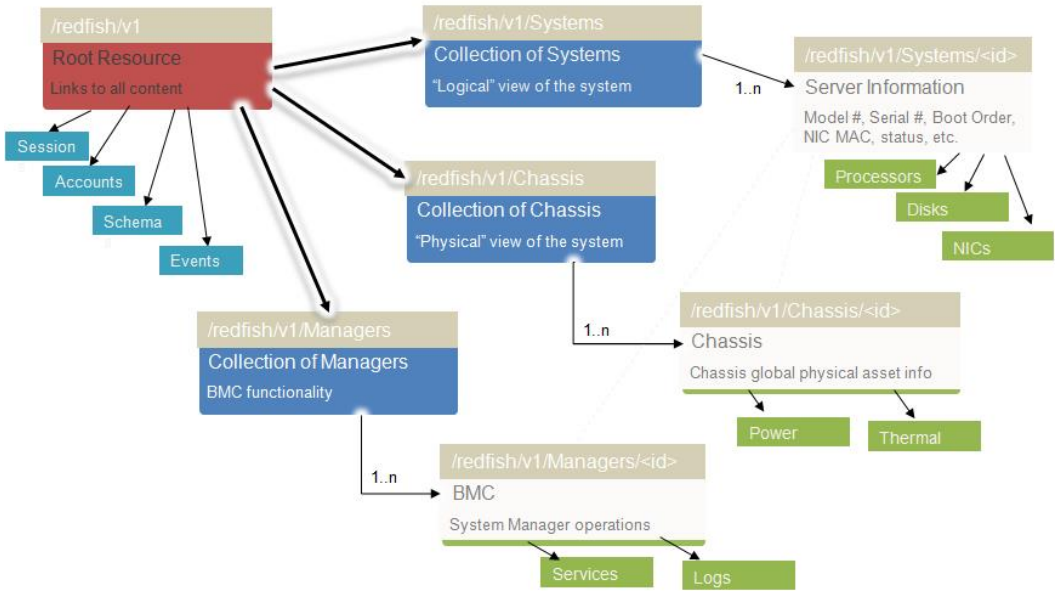
REST提出的设计概念和准则有：

- 网络上的所有事物都被抽象为资源，以JSON格式表示。
- 每个资源对应一个唯一的资源标识URI。
- 通过通用的HTTP接口（GET/POST）对资源进行操作。
- 对资源的各种操作不会改变资源标识。
- 所有的操作都是无状态的（stateless）。

Redfish可扩展平台管理编程接口，是一个管理标准，它基于HTTP协议，使用内置于超媒体RESTful接口的数据模型展现。

Redfish = REST API + 软件定义的服务器(数据模型)，当前由标准组织DMTF（www.dmtf.org）负责维护。

**图 2-2** Redfish Schema 框架




## 2.1.4 CLI 管理接口

CLI是openUBMC提供的一个私有命令行接口，包含两个基本命令程序：ipmcget和ipmcset，通过这两个命令程序就能实现对服务器的远程管理。可通过SSH方式登录openUBMC后执行此命令。

CLI接口不仅提供了不依赖额外工具的人机操作界面，也能用于被集成，比Web更轻量，比部分集成接口更友好。

## 2.1.5 Web 管理接口

openUBMC提供了基于HTTPS的Web可视化管理接口，使用户可以：

- 通过简单的界面操作快速完成设置和查询任务。
- 通过远程控制台可以对服务器进行OS启动全程监控、OS操作、以及光驱映射等。

可以在浏览器地址栏输入openUBMC的网口IP地址（IPv4或IPv6）或域名称打开openUBMC Web的登录界面，输入本地账号登录或LDAP域账号到openUBMC Web。

Web接口支持的OS和浏览器如表2-2所示。

**表 2-2** 客户端环境要求

| 操作系统 | 浏览器 |
| --- | --- |
| Windows 7 32位<br>Windows 7 64位 | • 支持Mozilla Firefox 78.0及以上版本，推荐99.0～101.0版本<br>• 支持Google Chrome 64.0及以上版本，推荐99.0～101.0版本 |
| Windows 8 32位<br>Windows 8 64位 | |
| Windows Server 2008 R2 64位 | |
| Windows Server 2012 64位 | |
| Windows Server 2012 R2 64位 | |
| Windows Server 2016 64位 | |
| Windows 10 64位 | • 支持Microsoft Edge 79.0及以上版本，推荐99.0～101.0版本<br>• 支持Mozilla Firefox 78.0及以上版本，推荐99.0～101.0版本<br>• 支持Google Chrome 64.0及以上版本，推荐99.0～101.0版本 |
| CentOS 7 | 支持Mozilla Firefox 78.0及以上版本，推荐99.0～101.0版本 |
| MAC OS X v10.7 | • 支持Safari 12.0及以上版本，推荐15.3～15.5版本<br>• 支持Mozilla Firefox 78.0及以上版本，推荐99.0～101.0版本 |

openUBMC支持更安全的TLS协议版本：

支持安全的TLS 1.2/1.3协议，TLS 1.3仅支持开启状态，TLS 1.2支持开启/关闭状态。TLS 1.2/1.3均默认开启。

# 2.2 虚拟 KVM 和虚拟媒体

通过远程控制台界面可以使用虚拟KVM、虚拟媒体和手动录像功能以及对系统上下电、重启操作；远程控制台支持HTML5技术，版本界面如图2-3。HTML5的远程控制台支持美式、日式、意大利键盘。

远程控制台支持工作在窗口模式和全屏模式，当处于全屏或分屏模式下，同时按下Ctrl Alt Shift组合键可弹出工具栏。

远程控制台支持退出后触发系统自动锁定，有效防止系统信息泄露或入侵。

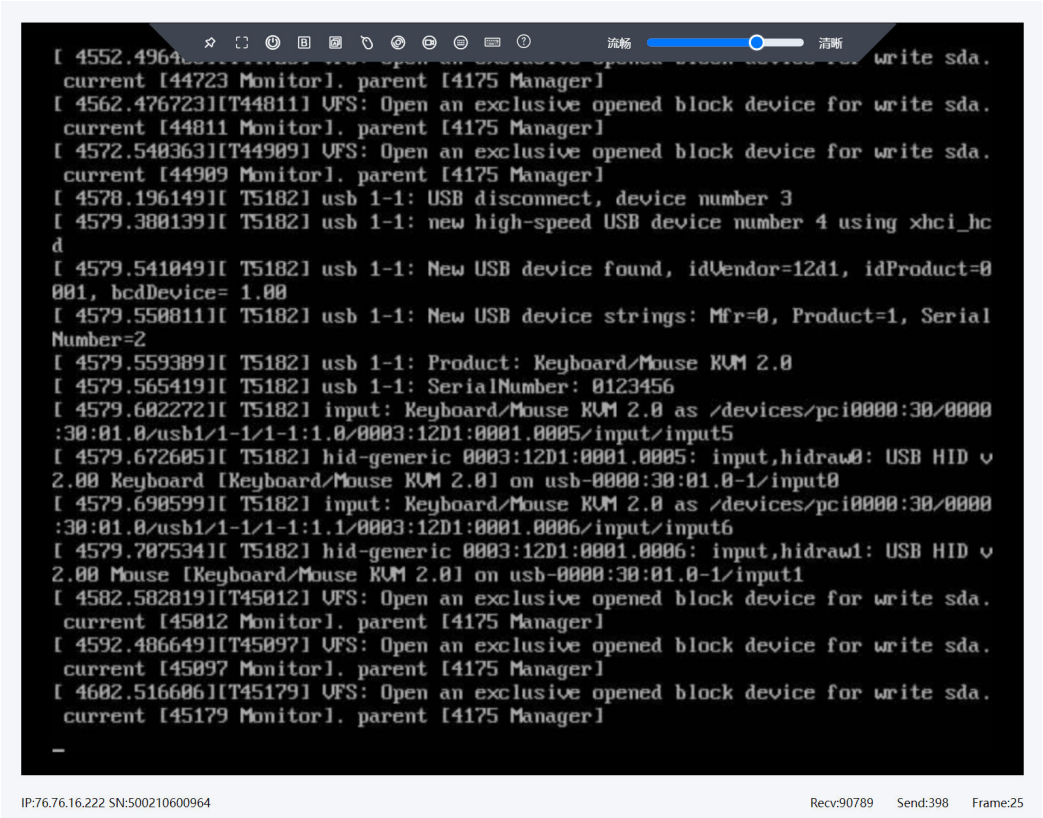
远程控制台支持启动方式如下：

1. VNC客户端启动，支持标准VNC协议及RealVNC、TightVNC、UltraVNC、TigerVNC四款主流VNC客户端。同时也支持通过Web首页HTML5 VNC客户端启动。
2. openUBMC Web和URL方式打开HTML5控制台，通过HTML JS加载控制台。

**表 2-3** 各种启动方式对比

| 分类 | 优点 | 缺点 | 备注 |
| --- | --- | --- | --- |
| 嵌入HTML5控制台 | 1、无需下载程序包，HTML JS加载。 | 1、对浏览器版本有要求。<br>2、不支持虚拟文件夹功能。 | |
| VNC控制台 | 1、标准协议，兼容第三方客户端。<br>2、无需安装JRE。 | 1、不支持虚拟媒体功能。<br>2、仅口令认证，无法按账号控制权限。 | |

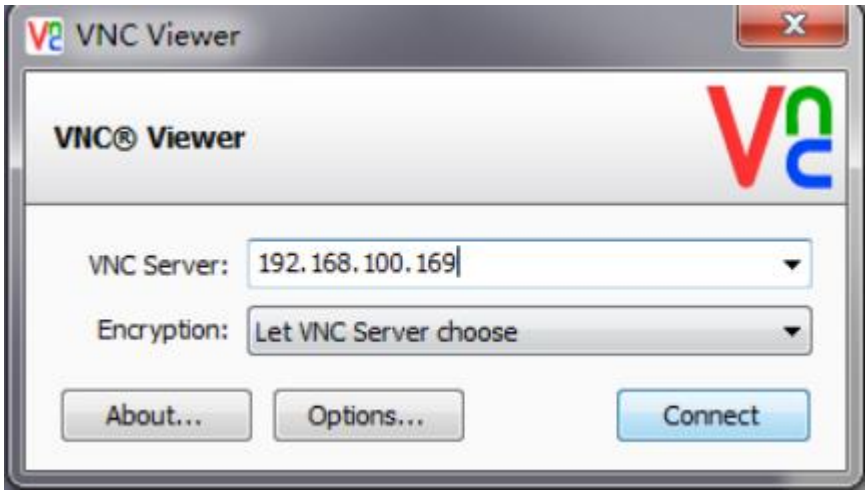
**图 2-3** HTML5 远程控制台（嵌入 Web）




**说明**

基于HTML5的控制台，无需安装额外软件，支持的浏览器版本：Microsoft Edge 79.0及以上、Firefox 78.0及以上和Chrome 64.0及以上。

**图 2-4** VNC 客户端（独立）




VNC协议具有如下特点：

1. VNC仅提供KVM功能，不支持虚拟媒体。

2. VNC遵循标准协议，能与第三方VNC客户端对接。
3. 仅提供密码认证，有自己独立的密码。
4. 使用跟键盘布局有关，支持美式键盘和日式键盘。

**图 2-5** VNC 配置界面

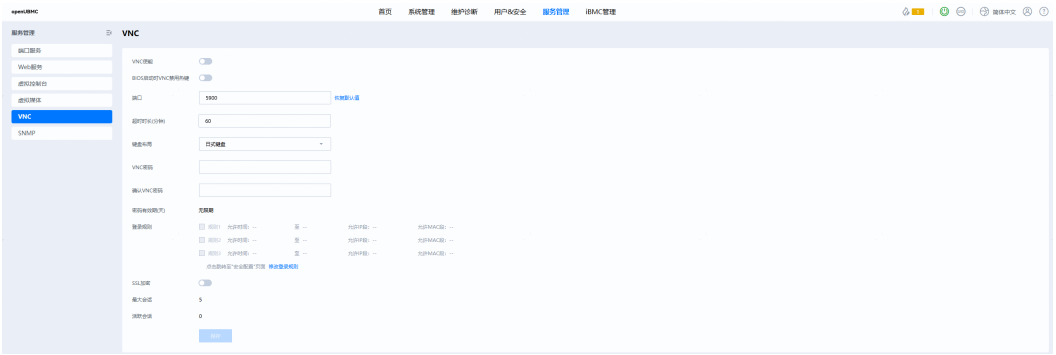
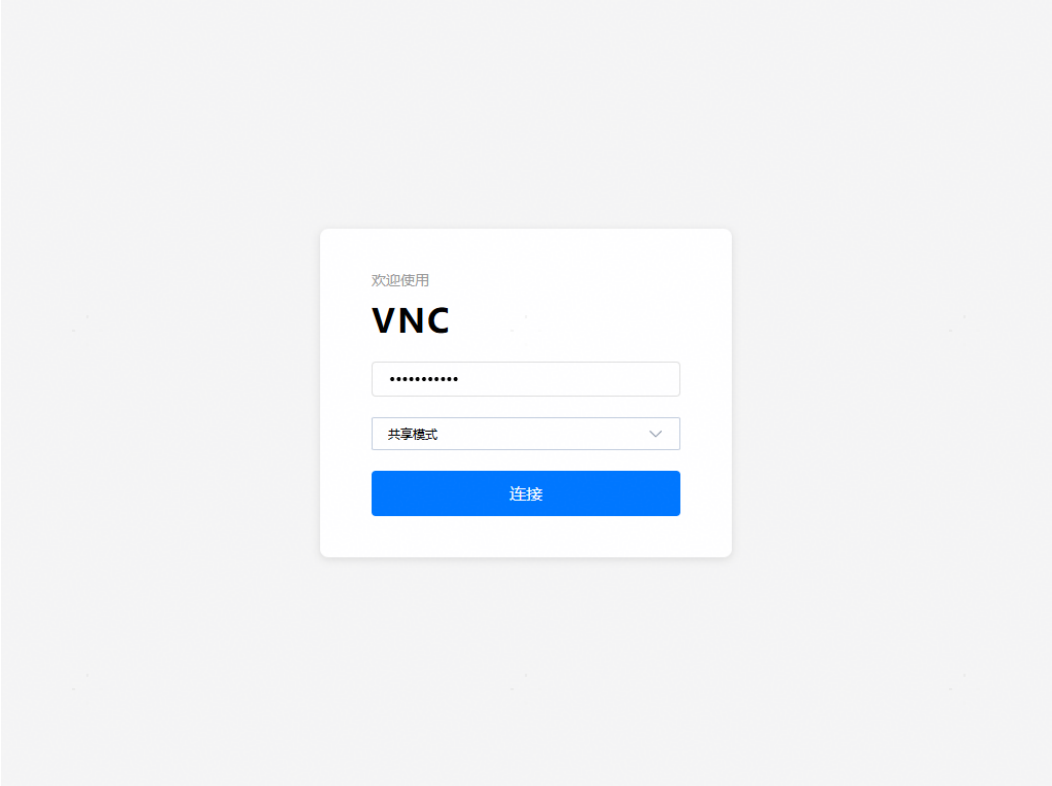
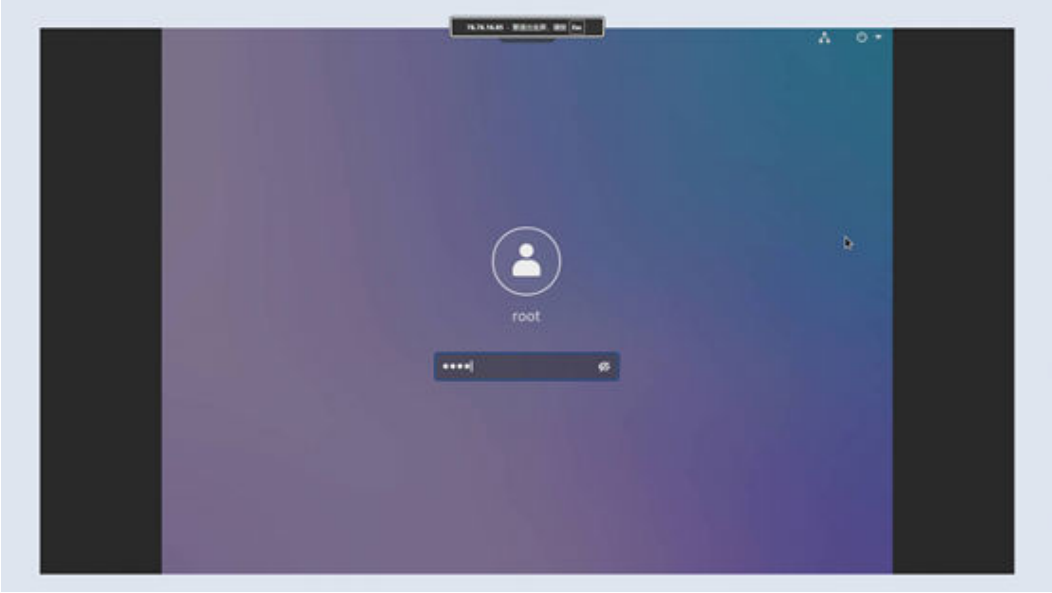



**图 2-6** HTML5 VNC 界面




## 2.2.1 虚拟 KVM

虚拟KVM是指用户在客户端利用本地的视频、键盘、鼠标对远程的设备进行监视和控制，提供实时操作异地设备的管理方式；主要特点如下：

- 分辨率：最高分辨率为1920*1280（实际能支持的最大分辨率跟OS有关），最低分辨率为640*480。
- 鼠标同步：远程服务器鼠标跟随本地鼠标移动，该功能需要远端服务器OS支持。
- 鼠标模式：支持绝对、相对和单鼠标三种模式。

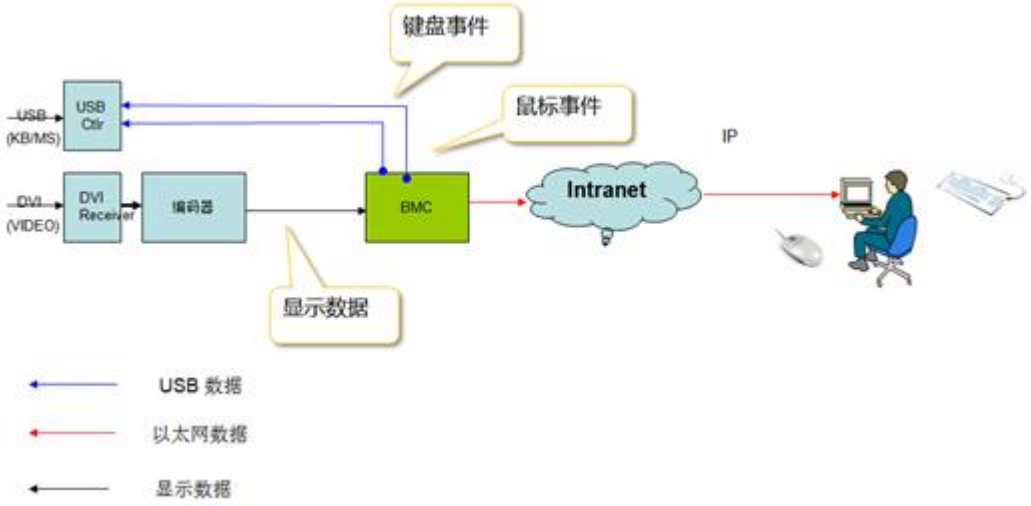
- 工作模式：支持独占和共享模式，共享模式下，协同双方可以同时操作远端服务器；独占模式下，同一时间只有一个会话。
- 运行环境：使用虚拟KVM功能，客户端需具备相应版本的浏览器、OS，如表2-2所示。
- 色彩位：支持32位真彩色，最多1677万种色彩。
- 组合键：支持5个默认组合键和用户自定义组合键。
- 加密：视频、键盘和控制命令数据支持TLS加密传输。

由于鼠标同步功能取决于OS是否支持提供绝对鼠标位置信息，所以对于不能提供绝对鼠标位置信息的OS，KVM不支持鼠标同步功能。

虚拟KVM的实现原理如图2-7所示：

- openUBMC将远端的显示数据压缩编码后通过网络传输到用户所在的客户端主机，由客户端主机控制台解码解压缩后恢复显示。
- 虚拟KVM的控制台会将用户所在的客户端主机的鼠标事件和键盘事件捕获，通过网络传输到远端，由openUBMC智能管理控制器模拟远端的键盘鼠标将事件经由USB通道输入到远端服务器业务系统上。

**图 2-7** 虚拟 KVM 实现原理




## 2.2.2 虚拟媒体

虚拟媒体即通过网络在服务器上以虚拟USB光盘驱动器和软盘驱动器的形式提供对本地媒体（光盘驱动器或镜像文件，硬盘文件夹）的远程访问方式；虚拟媒体数据支持AES 128 CBC算法加密传输。使用虚拟媒体功能，客户端需具备相应版本的操作系统如表2-2所示。

虚拟媒体的实现原理是将客户所在的本地主机的媒体设备通过网络虚拟为远端服务器主机的媒体设备，如图2-8所示。


**图 2-8** 虚拟媒体实现原理

openUBMC与服务器主机的数据通道采用USB2.0协议。目前openUBMC的虚拟媒体具有以下功能特性：

- 虚拟设备

  虚拟设备即将客户端的PC设备或者镜像文件映射到建立连接的服务器上，使得该服务器检测到一个USB设备。

  虚拟设备包括如下多种情况：

  –虚拟一个光驱设备

  –虚拟一个文件夹，包括本地和网络上的文件夹
- 虚拟媒体性能

  –虚拟光驱支持的最大传输速率为32 Mbit/s，VLAN时支持的最大传输速率为24 Mbit/s

# 2.3 基于 HTTPS 的可视化管理接口

openUBMC提供了基于HTTPS的Web可视化管理接口，可以实现通过简单的界面操作快速完成设置和查询任务，支持的具体浏览器和OS版本如表2-2所示，以下图示以2280产品为例，其它不同形态产品的界面可能存在差异。Web界面支持中文、英文两种语言，并支持在两种语言之间切换，默认与浏览器的语言一致。

可按照如下方式登录openUBMC Web：

**步骤1** 在浏览器URL地址栏输入*https:// openUBMC IP[:sslport]*，如图2-9所示。

**说明**

端口号是可选的，若port不为80或sslport不为443则IP地址后面必须要带上端口号，端口号修改方法参考图2-10。

**图 2-9** 输入 openUBMC 地址

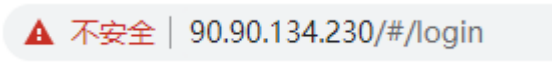
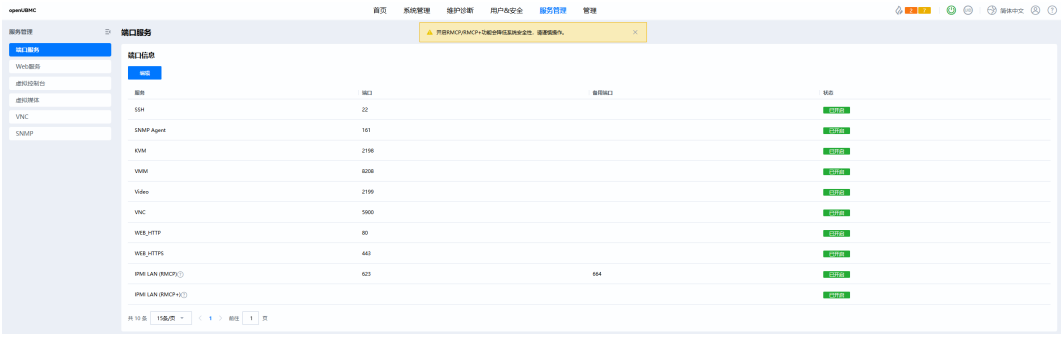
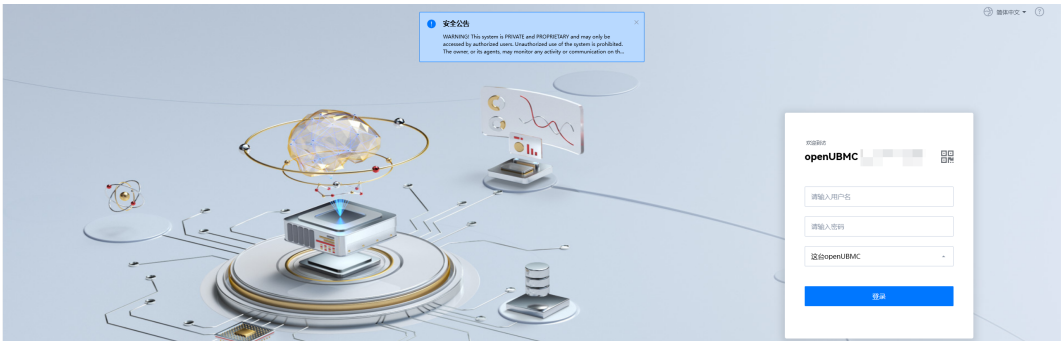



**图 2-10** 端口修改方法



**步骤2** 在用户登录界面中输入用户名和密码，若是域账号登录则选择登录到具体的域，然后单击下方的“登录”按钮登录，如下图所示。

**图3** 登录openUBMC Web




**----结束**

## 2.3.1 查看系统总体概况

总体概况界面显示系统当前基本情况，包括系统状态、openUBMC信息、系统配置信息、虚拟按钮和虚拟控制台链接信息，并提供常见操作接口链接，如图2-11所示

**图 2-11** 总体概况界面

## 2.3.2 查看系统信息

系统信息界面详细显示当前系统的固件版本、资产信息和整机硬件信息。

### 固件版本

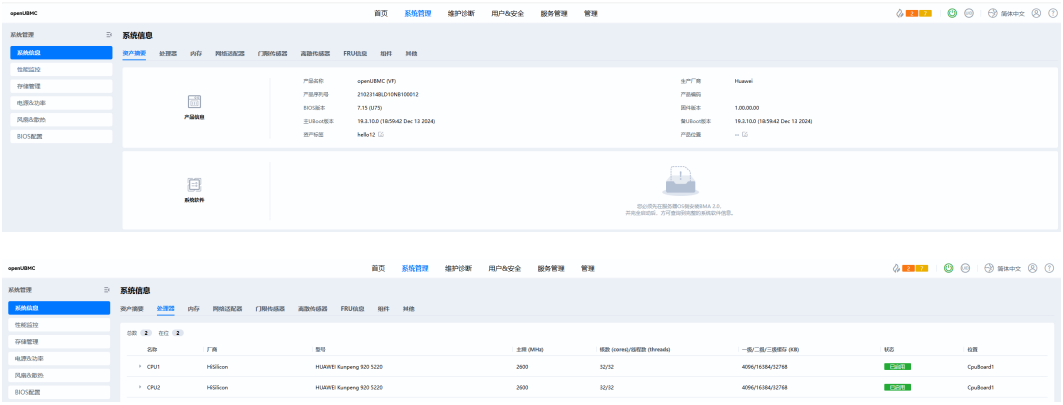
固件版本包括openUBMC固件、BIOS固件、以及型号、主机名、全局唯一标识符、序列号、MAC地址，如图2-12所示。

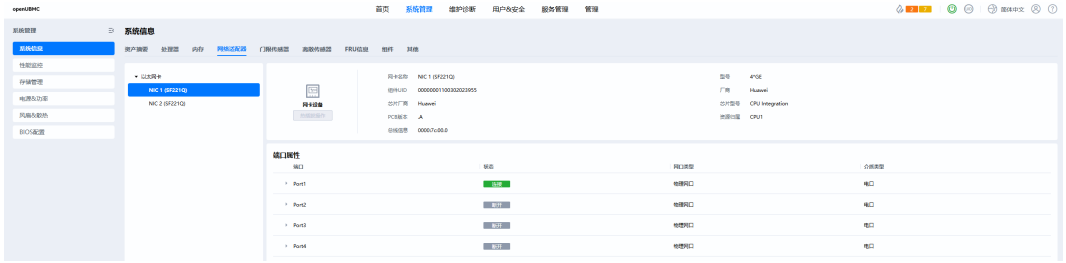
**图 2-12** 固件版本界面

### 整机硬件

整机硬件信息包括系统主要部件的最大配置数、当前配置数和型号，其中“网络”和“系统软件”部分需要安装BMA2.0软件，如下图所示。

**图 2-13** 整机硬件界面





## 2.3.3 性能监控

实时监控主要是传感器信息和操作接口。

### 传感器

传感器界面显示设备所有传感器信息，如图2-14所示，相关的参数如表2-4所示。

**图 2-14** 传感器界面示例

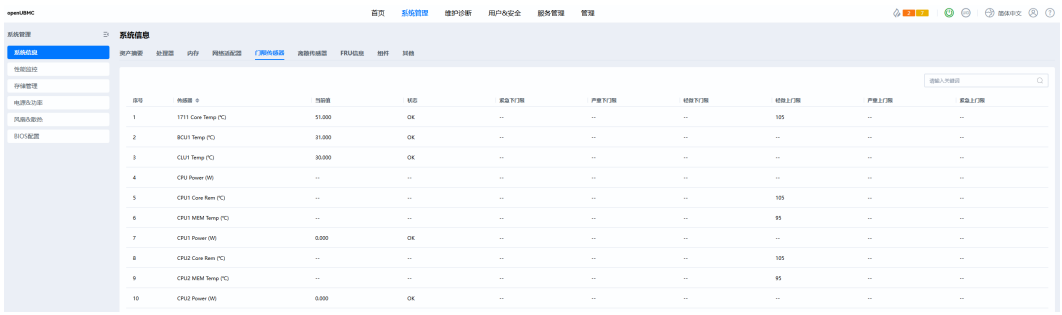


**表 2-4** 门限传感器界面各参数说明

| 参数 | 说明 |
| --- | --- |
| 传感器 | 传感器的名称。 |
| 当前值 | 传感器的当前值。 |
| 状态 | 传感器的状态。 |
| 紧急下门限 | 传感器值低于此下限值时，系统会产生紧急告警。 |
| 严重下门限 | 传感器值低于此下限值时，系统会产生严重告警。 |
| 轻微下门限 | 传感器值低于此下限值时，系统会产生轻微告警。 |
| 轻微上门限 | 传感器值高于此上限值时，系统会产生轻微告警。 |
| 严重上门限 | 传感器值高于此上限值时，系统会产生严重告警。 |
| 紧急上门限 | 传感器值高于此上限值时，系统会产生紧急告警。 |

## 2.3.4 设备定位

如图2-15所示，在设备定位界面，可以根据实际需要设置定位指示灯状态，通过点亮定位指示灯，使用户可以在机房的大量设备中，快速定位到需要执行现场操作的设备。

**图 2-15** 设备定位界面

# 2.4 域管理

随着企业应用的发展，IT基础架构的容量也越来越大，带来的资产管理和日常管理工作量也呈数量级增长。为了应对越来越繁重的IT基础架构管理工作，openUBMC智能管理系统提供了域管理。

## 2.4.1 域管理

用户可以将所有被管理服务器加入一个统一的管理域并使用域名来访问被管服务器的openUBMC。如果在加入域的同时使用被管服务器的资产编号作为域名，还可以通过域控制器实现自动资产盘点，大大降低IT资产管理的成本。

**步骤1** 加入域。

1. 在openUBMC的Web中打开“网络配置”界面，如图2-16所示。

   **说明**

   DNS（Domain Name System）是因特网的一项核心服务，将域名和IP地址相互映射，使用户可以通过域名直接访问网络，而不必去记住对应的IP地址。

2. 在图2-16中，用户可以配置DNS绑定网口及DNS信息获取模式。设置完毕后单击“保存”执行操作。
3. 当用户选择“手动配置DNS信息”时，需要同时配置域名以及相应的首选、备用DNS服务器。

**图 2-16** DNS 配置界面

**步骤2** 在如图2-17所示界面中设置主机名。

**图 2-17** 主机名配置界面

**----结束**

# 2.5 固件管理

openUBMC可管理的固件包括openUBMC固件、BIOS、CPLD、CSR（Component Self-Description Record）、电源、MCU等，支持固件版本查询、固件升级、BMC双镜像切换。

## 2.5.1 固件双镜像

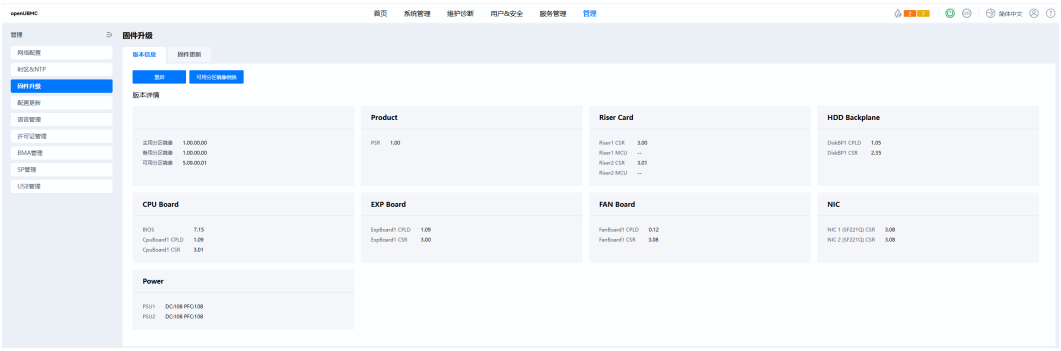
为了提升系统可靠性，openUBMC使用了固件双镜像备份技术。当在网运行过程中出现flash误操作或者存储块损坏时，系统会自动切换到备份镜像运行。

### 通过 Web 切换镜像

在导航树上选择“系统管理 > 固件升级”，打开“固件升级”界面。如图2-18所示。

在固件版本视图窗口中，显示openUBMC固件和相关固件的当前版本信息，并可进行镜像切换和重启openUBMC操作。

**图 2-18** 固件升级界面




## 2.5.2 固件升级

支持对openUBMC固件、BIOS、CPLD（基础板/背板/扩展板）、电源、MCU等固件的升级；其中openUBMC固件支持主备镜像倒换回滚和本地固件更新，如图2-19所示。从兼容性考虑，建议用户将openUBMC主备镜像更新到同一个版本。

**图 2-19** 固件升级界面

## 2.5.3 BMC 升级与生效分离

基于Hi1711芯片的openUBMC，支持openUBMC固件升级与生效分离的能力，默认升级后立即复位生效，支持用户选择下次复位时生效。

## 2.5.4 固件并行升级

勾选"并行升级模式"的情况下，支持对BMC/BIOS/CPLD/MCU等固件进行并行升级，提高升级效率（并行升级任务最大数量为20个）。上电情况下，升级完CPLD/BIOS等固件后，在"启动生效"页面触发固件生效。

## 2.5.5 升级固件是否保留配置选择

用户可以选择在升级BMC/BIOS固件后，保留或者清除BMC/BIOS配置，默认行为是升级后保留配置。

## 2.5.6 固件恢复

当BMC/BIOS/CPLD升级被意外中断（如发生整机掉电），导致固件异常时，能够自动修复，免人工干预，提升系统整体韧性。

## 2.5.7 固件整包升级

用户可以升级整包固件，整包固件为多个.hpm固件压缩为一个.zip的压缩包，可以解决用户取包效率和固件包的版本配套问题，如图1整包固件升级所示。整包固件升级支持用户二次选择，可以选择部分固件或全部固件启动升级，如图2整包固件二次选择。

图1 整包固件升级

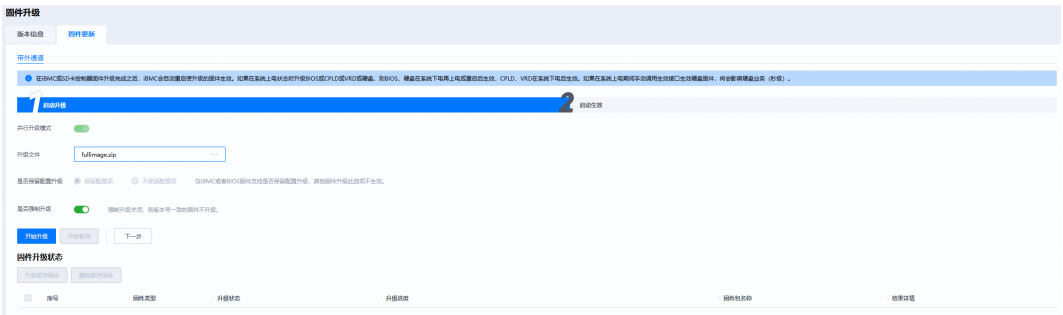
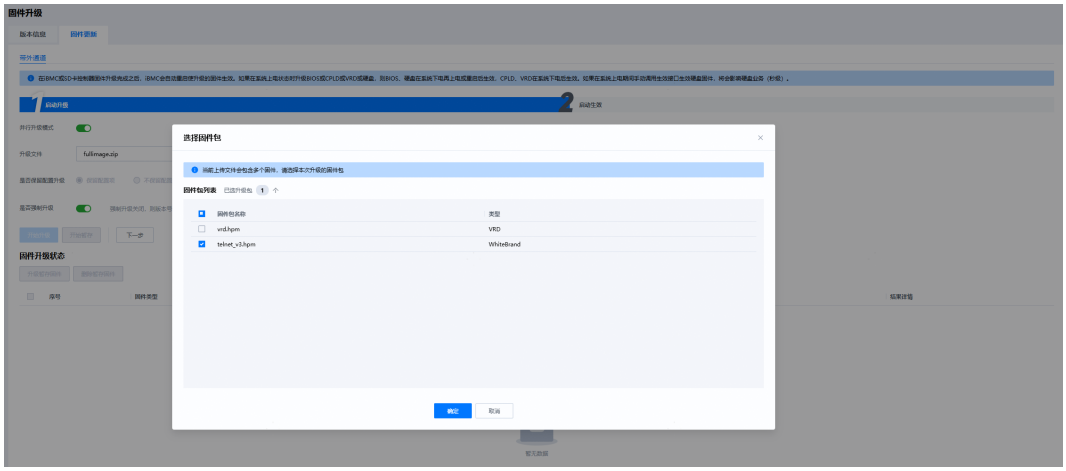


图2 整包固件升级二次选择




## 2.5.8 固件暂存升级

用户可以暂存固件，即将固件包存在BMC flash中，暂不进行升级和生效，图1为固件暂存按钮。固件暂存后可以支持查询暂存的固件，并且支持选中对应固件后进行“升级暂存固件”或“删除暂存固件”。

图1 固件暂存接口


图2 暂存固件查询、升级、删除接口

## 2.5.9 CPLD 无感升级

新增CPLD无感升级场景，允许CPLD在正常运行过程中上传固件包，执行升级操作并立即生效。为确保此时CPLD业务功能不受影响，无感升级流程中BMC需要与CPLD进行通信配合，提前告知业务上使用CPLD的消费模块即将开始无感升级，需要做好升级准备；同时BMC需要负责CPLD升级完成后的关键寄存器内容恢复等。为实现以上无感升级功能，BMC需要完整开发无感升级分支，正确识别升级流程中上传的固件包是否为无感升级包，并在无感升级过程中做好必要的交互和恢复现场措施，确保无感升级不影响业务功能。

主要功能点如下：

1. 解析上传的固件包信息，判断是无感升级固件包还是冷升级固件包；
2. 获取固件包中的cpld.svf文件和valid.svf文件，使用底层链路接口将这些文件写入CPLD，完成升级与生效动作；
3. 根据CSR配置决定是否需要进行无感升级前置处理，决定是否需要通知CPLD逻辑数据持久化；
4. 根据CSR配置，访问CPLD一组给定地址的寄存器，将寄存器值保存下来，用于后续升级生效后恢复；
5. 根据CSR配置，向特定的CPLD寄存器中写入标记位，通知各消费模块即将开始无感升级，并读取一组给定的寄存器，等待各消费模块返回无感升级前的准备结果。同样在无感升级完成后使用这些寄存器通知消费模块升级已完成；
6. 根据CSR配置，操作无感升级流程专用GPIO，升级前拉低该GPIO，升级后拉高该GPIO。

**图 2-20** CPLD 无感升级界面

**图 2-21** CPLD 无感升级成功版本号刷新


## 2.5.10 网卡带外升级

一、带外升级

支持私有 smbus 命令进行网卡固件的升级和激活，实现 1825网卡固件升级的完整生命周期管理，包括升级信号监听、固件数据传输、升级状态查询、预生效流程，提供稳定、可靠的网卡固件升级能力。

1. 升级信号监听
   - 监听网卡固件升级请求，触发对应升级流程处理。
2. 固件数据传输
   - 通过私有 SMBus 命令完成待升级固件的分片传输，每片 1.5KB，每帧 256B；
   - 传输规则：多个子固件按顺序升级，每个子固件先发送固件头，再发送固件数据，通过 sub_type 区分类型。
3. 升级状态查询
   - 支持子固件传输完成后的 Flash 写入状态查询；
   - 支持超时重试机制。
4. 预生效与激活
   - 支持发送固件预生效命令；
   - 待 OS 复位联动网卡复位后，自动激活新固件。

二、并行升级

现有网卡固件升级采用串行升级方式，逐个网卡顺序执行升级，多网卡场景下升级效率低下，整体升级耗时随网卡数量线性增长。针对SMBus协议升级引入并行升级机制，利用skynet协程并发执行多网卡升级任务，限制最大并行数，在提升升级效率的同时避免资源过载，串行升级基础上，为1825网卡SMBus升级提供并行升级能力。

1. 升级信号监听
   - 监听网卡固件升级请求，当已存在升级任务时，直接报错返回
   - 判断是否为smbus升级，触发对应升级流程处理
2. 并行升级流程
   - 网卡升级对象需要新增属性LockChip用于升级过程的总线加锁和判断
   - 解析 update.cfg 文件获取 componentID 和 componentIDEx
   - 根据 componentID 和 componentIDEx 匹配获取可升级网卡列表
   - 创建并发调度器，容量为最大并行数，默认为4
   - 通过LockChip进行总线占用的判断和加锁
   - 通过skynet.fork为每个网卡创建独立升级协程，skynet.wakeup/sleep 机制保护，确保线程安全：
     - 协程获取并发槽位，槽位已满则进入等待队列（ FIFO ）
     - 调用 SMBus 接口执行单个网卡固件升级
     - 升级完成后释放槽位，唤醒等待队列中的下一个协程
   - 主协程等待所有升级任务完成
   - 统计升级成功/失败数量，返回升级结果

**图 2-22** 网卡固件升级界面


**图 2-23** 网卡固件升级成功界面


# 2.6 电源管理

为了降低运营TCO，openUBMC智能管理系统提供了多种电源管理功能。

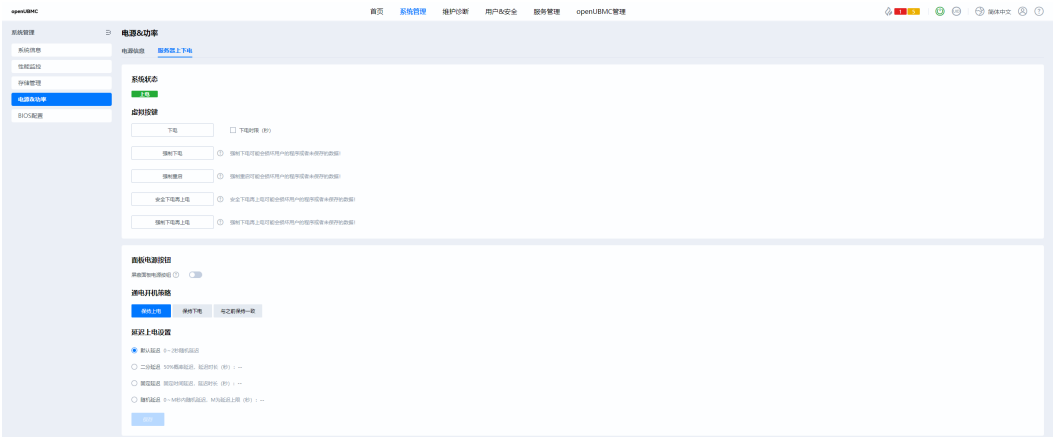
## 2.6.1 电源控制

电源控制界面提供对服务器的电源控制方式，如图2-24所示。

服务器电源控制方式包括：上电、下电、强制下电、强制重启、强制下电再上电。

- 上电：表示对服务器进行上电。
- 下电：表示对服务器进行安全下电，openUBMC向OS发送ACPI中断，若OS支持ACPI服务，则先走正常的操作系统关闭(将所有运行进程关闭)后下电，否则，只能等到超过下电超时时间后，openUBMC将系统强制下电；效果相当于短按服务器面板上的电源按钮。
- 强制下电：表示对服务器进行下电，无需等待OS响应，绕过正常的操作系统关闭流程，效果相当于长按服务器面板上的电源按钮。
- 强制重启：表示对服务器进行冷复位，绕过正常的操作系统关闭流程。
- 强制下电再上电：表示对服务器先安全下电再上电，实现按序重启，即：先走正常的操作系统关闭流程并下电，若设置的安全下电超时时间内不能完成下电则强制下电，最后再上电。

**图 2-24** 电源控制

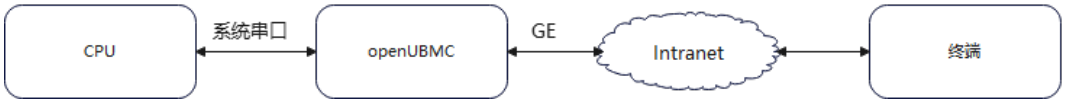


# 2.7 系统串口重定向及运行记录

## 2.7.1 系统串口重定向

openUBMC提供系统串口重定向(SOL：Serial Over LAN)功能，即将原本只能从近端串口线输出的系统串口数据重定向到网络设备输出，并能接受远程网络设备的输入。支持IPMI SOL和命令行SOL两种方式，但这两种方式互斥，其中命令行SOL支持同时打开两个SOL会话。如图2-25所示原理，网管人员在远程通过网络终端就可以轻松的查看系统串口实时输出数据，并能对系统进行操作干预，跟在近端使用系统串口一样的效果。

**图 2-25** 系统串口重定向原理




## 2.7.2 系统串口信息记录

openUBMC提供系统串口信息记录功能。如图2-26所示原理和展示方式，系统串口信息记录将系统串口的实时数据记录到DDR中，循环覆盖，最多保留最近2M字节的系统

串口数据；当系统发生宕机或重启故障时，可以从openUBMC导出信息记录并查看详细的信息。

**图 2-26** 系统串口信息记录原理

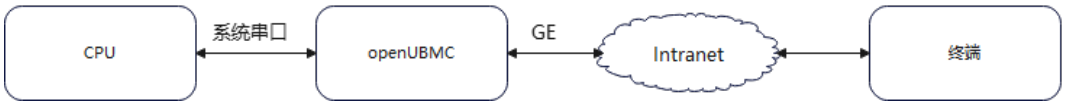



**图 2-27** Web 展示界面


# 2.8 账号与身份认证

openUBMC是一个基于嵌入式CPU和OS的管理子系统，OS和应用对外是一个封闭的整体，只提供了固定的维护、集成接口。OS(CLI)、SNMP、IPMI LAN、WEB、redfish等这些对外接口各自都有一套独立的本地用户管理，对用户来说，要想通过这些接口都能接入，则必须重复五遍配置用户的动作，非常繁琐。因此，我们提供了统一用户管理的功能，只要在上述任一接口配置好用户，即可使用该用户登录openUBMC所有接口，也就是说所有接口呈现的本地用户是同一套；openUBMC后台自动完成了各个接口的用户同步。

本地用户最多支持16个用户，支持增加、修改和删除用户；所有用户划分为管理员、操作员和普通用户三个固定权限组和一个自定义权限组，每个组的具体权限如下：

管理员：拥有openUBMC的所有配置和控制权限

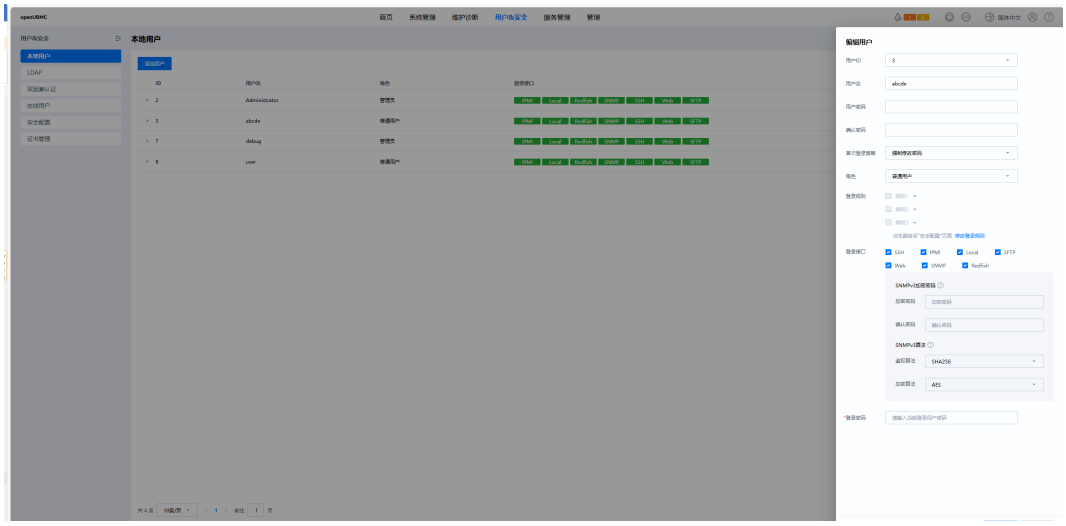
操作员：相对于管理员，拥有除用户管理和安全配置外的所有配置和控制权限

普通用户：只有查看权限，除OS相关信息和操作日志查看外的所有查看权限

自定义组：由用户指定该组的具体权限

登录接口：由创建者指定新用户可以使用的接口类型

**图 2-28** 用户管理界面



## 2.8.1 账号

服务器带外管理软件openUBMC支持CLI、SNMP、Web、IPMI、Redfish等管理接口，并提供了统一的用户管理功能。最多支持16个用户，支持增加、修改和删除用户。

账号安全包括：密码复杂度检查、弱口令字典、禁用历史密码、密码有效期、密码最短使用期、不活动期限、紧急登录用户、账号防暴力破解（登录失败锁定）、账号手动锁定、在线用户注销。

### 2.8.1.1 用户配置策略

- **账号防暴力破解：**账号支持基于用户连续多次登录失败锁定，及SNMP超长团体名的防暴力破解机制；
- **登录失败锁定：**支持登录失败次数，锁定时间的配置；当用户连续输入错误密码的次数超过设置的“错误次数”时，该用户被锁定。用户被锁定后，在锁定时长内不能继续登录，可以通过管理用户登入命令行手工解锁。如不进行手动解锁，系统会在超过锁定时间时自动解锁。
- **SNMP超长团体名：**启用SNMP超长团体名后，设置的团体名必须大于等于16个字符，团体名设置也支持复杂度检查，防止设置简单团体名带来的风险；
- **紧急登录用户：**不受密码有效期、登录规则、登录接口限制的用户，用于紧急情况下登录openUBMC，默认为空。
- **不活动期限：**超过设定期限内未活动的用户会被禁用。

### 2.8.1.2 密码配置策略

**密码复杂度检查：**对用户配置的密码的复杂度进行校验，避免用户设置过于简单的密码。用户可以自行设置密码复杂度规则，默认密码复杂度要求：

- 长度为8 ~ 20个字符。
- 至少包含一个空格或者以下特殊字符： `~!@#$%^&*()-_=+\|[{}];:'",<.>/?
- 至少包含以下字符中的两种： 小写字母：a ~ z； 大写字母：A ~ Z；数字：0 ~ 9

- 不能是用户名或用户名的倒序。
- 新旧口令至少在两个字符位上不同。

**禁用历史密码：**支持用户配置保留历史密码的个数，设置的新密码不允许和历史密码相同。

**密码有效期：**支持用户配置密码有效期时间，密码达到有效期后必须修改新密码才能登陆；密码有效期小于10天时，系统会提示用户修改密码。

**密码最短使用期：**设置一个密码后，要使用的最短时间，在此时间内不能修改密码；设置密码最短使用期的目的在于防止频繁修改密码而重复使用历史密码的风险，确保密码安全。

**弱口令字典：**支持CLI接口设置弱口令字典认证使能状态和导入、导出弱口令字典，在密码复杂度检查和弱口令字典认证功能使能的情况下，所设置密码不能在弱口令字典中。

**基于接口的分权分域的用户管理：**串口、ssh只允许特定用户登录，且串口用户可以修改ssh用户密码、ssh用户不能修改串口用户密码。

## 2.8.2 认证

用户和上层管理系统通过Web、CLI、SNMP、IPMI、Redfish接口对openUBMC的访问都需要进行认证，用户口令采用PBKDF2算法计算口令单向哈希，可有效防止密码被明文破解。认证通过后才能进行设备的管理配置和信息查询等操作。

openUBMC支持本地认证、LDAP两种模式。支持“用户名 + 密码”认证、SSH 公钥认证、双因素认证以及重要操作的二次认证。

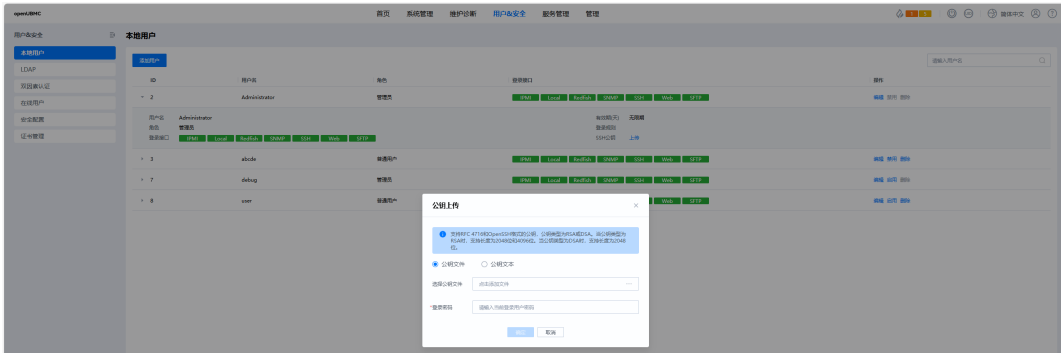
**SSH公钥认证：**SSH支持用户名、密码和公钥方式认证，公钥方式适合于自动配置工具，无需输入密码的交互步骤。

SSH公钥认证有如下优点：

- 登录验证时无需交互密码
- 密钥长度很长，不容易被人偷窥或猜测到

支持RFC 4716和OpenSSH格式的公钥，公钥类型为RSA或DSA。当公钥类型为RSA时，支持长度为2048位和4096位；当公钥类型为DSA时，支持长度为2048位。

每个账号只支持配置一个公钥，公钥导入支持文本输入和文件导入，导入后可查看该公钥的哈希值。基于更多安全考虑，启用SSH公钥认证后可禁用SSH的密码认证方式。



**双因素认证：**双因素认证是使用客户端证书密码以及证书来进行认证，登录时需要同时拥有客户端证书及证书密码才能认证通过，解决了传统的账号口令认证中口令泄露

导致的入侵问题。双因素认证开启后，只有客户端证书被openUBMC中导入的CA根证书验证通过，且跟导入到openUBMC中的客户端证书一致，才允许登录，当前只有WEB支持双因素认证。双因素认证开启后不支持基于用户口令、LDAP的认证，主要特性如下：

- 最多支持导入32个不同的CA根证书；
- 开启双因素认证后，会关闭Redfish、SSH接口，保留SNMP、IPMI接口跟网管软件对接；双因素认证功能默认关闭，可以通过Web、SNMP接口配置开启；
- 支持证书吊销检查，默认关闭，吊销检查开启后，已被吊销的证书不允许登录。


二次认证：对于重要的管理操作，如用户配置、权限配置、公钥导入会对已登录用户进行二次认证，认证通过后才能执行重要操作，防止用户登录后没有断开链接，被其它非法用户执行恶意操作。

### 2.8.2.1 访问策略

- 登录规则

  支持基于场景的登录限制，基于时间段、IP、MAC的访问控制策略，通过配置登入时间段、登入IP网段、登入MAC地址白名单，只允许满足白名单要求的用户通过管理通道访问系统，对系统进行管理和配置，将服务器管理接口访问控制在最小范围；

  由用户根据需要设置登录规则的白名单，最多支持三条登录规则，登录时只要匹配上任意一条登录规则，即可登录，否则拒绝登录；

  每条登录规则包括时间段、登录用户的源IP段和MAC段，这三个条件必须同时满足才认为匹配到一条登录规则；登录规则可应用于所有本地用户和LDAP用户组；

  三维立体象限：

  时间段：包括开始时间和结束时间，时间格式必须一致，支持YYYY-MM-DD HH:MM、YYYY-MM-DD和HH:MM三种格式，允许为空；

  IP段：支持单个IPv4地址或IPv4地址段，允许为空，目前不支持IPv6地址；

  MAC段：支持单个MAC地址或MAC地址段，允许为空。


  登录规则应用场景：

  时间段：只在特定的时间段允许登录维护，比如有些数据中心下班后不允许登录操作，就可以通过配置登录时间来进行控制，以降低安全风险。

  IP段、MAC段：只允许特定范围内的IP、MAC才能登录，防止网络上的大规模异常攻击。

- 系统锁定

  支持系统锁定功能，系统锁定功能开启后，系统中的用户配置、常规配置、虚拟控制台配置、安全配置都处于锁定状态不能配置，系统电源控制、虚拟媒体功能和查询功能可以正常使用。系统锁定功能可以防止系统配置的意外或恶意更改。

  只有管理员权限用户才有系统锁定功能开启和关闭的权限，开启后，WEB、CLI、SNMP、Redfish、IPMI接口都被锁定，无法进行配置。

## 2.8.3 授权

- openUBMC中用户划分为管理员、操作员、普通用户和自定义用户等权限组，每个组的具体权限如下：
  - 管理员：拥有的所有配置和控制权限。
  - 操作员：相对于管理员，拥有除用户管理、调试诊断和安全配置外的所有配置和控制权限。
  - 普通用户：只有查看权限，除OS相关信息和操作日志查看外的所有查看权限，并能修改自身密码。
  - 自定义权限组：自定义权限组允许系统管理员根据用户的实际场景自定义精确分配用户权限。openUBMC支持最大4个自定义权限组。系统权限类型被划分为用户配置、常规设置、远程控制、远程媒体、安全配置、电源控制、调试诊断、查询功能、配置自身这几种类型，系统管理员可以灵活将这些权限类型配置为一个自定义权限组。

**图 2-29** 自定义角色应用

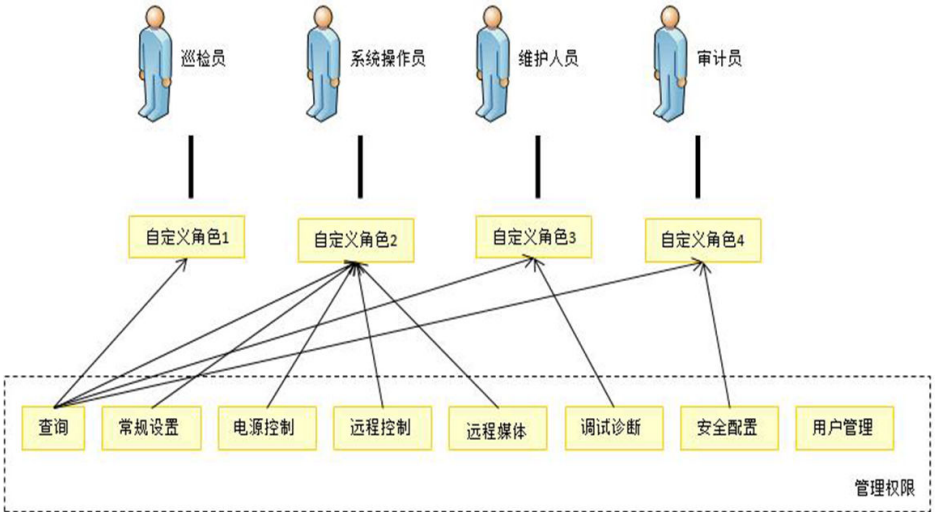
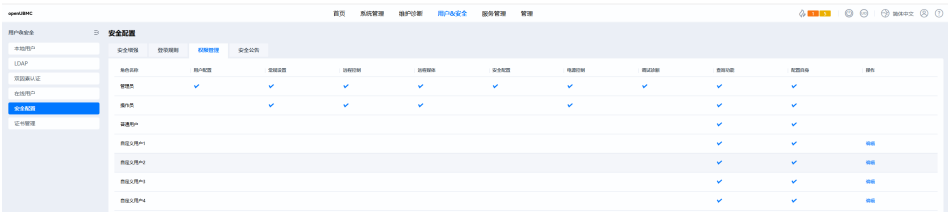



**图 2-30** 角色自定义界面



## 2.8.4 会话

**会话建立与认证：**支持本地帐号、LDAP两种认证源，并可集成双因素认证。认证通过后，系统为每个会话分配唯一的会话标识(Session ID)，会话标识使用安全随机数生成，并在后续交互中进行严格校验。

**会话销毁：**有两种方式终止会话，

1）超时终止：对于CLI、Web、SFTP等长连接会话实现了静默超时断连机制，超过超时时间没有操作则会自动断开会话。

2）手动终止：用户主动发起请求终止当前会话。另外，管理员可实时查看当前活跃会话列表，管理员可以主动终止其它会话。


**并发会话控制：** 可以为不同类型会话定义最大并发会话数，确保关键管理员帐号不会因过多并发登录而被滥用或拦截。

**在线用户：**支持查看已登录openUBMC系统的用户信息，并支持注销已登录的用户。

**会话销毁：**有两种方式终止会话，

1）超时终止：对于CLI、Web、SFTP等长连接会话实现了静默超时断连机制，超过超时时间没有操作则会自动断开会话。

2）手动终止：用户主动发起请求终止当前会话。另外，管理员可以主动终止其它会话。

## 2.8.5 目录服务

按照如图2-31所示原理，启用iBMC的目录服务，可以将所有iBMC的用户管理，权限分配，有效期管理都集中到目录服务器上，避免大量的重复性用户配置任务，提高管理效率。另外将用户集中到目录服务器上，也能大大提高iBMC智能管理系统的安全性。

LDAP标准优点：

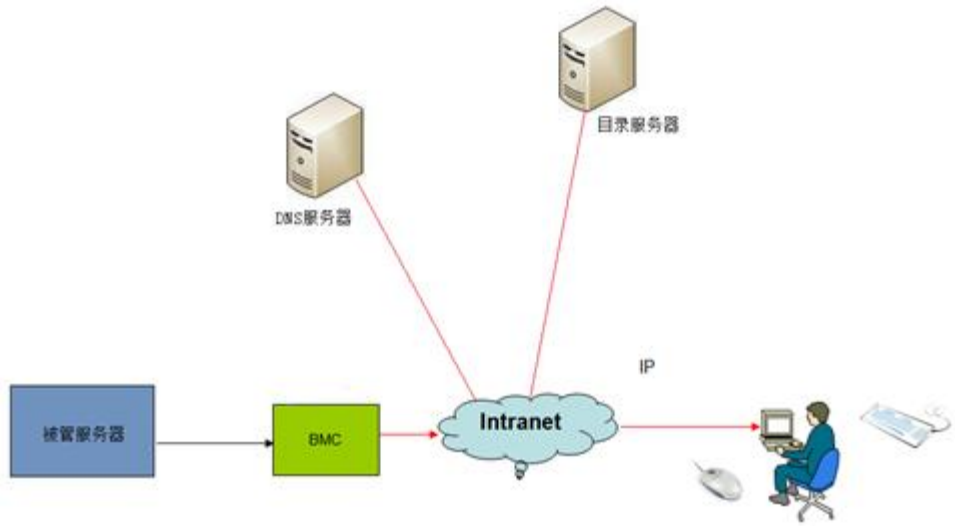
1. 可扩展性：可以在所有iBMC上同时动态支持LDAP服务器上新增账户的管理。
2. 安全性：用户密码策略都在LDAP服务器上实施。
3. 实时性：LDAP服务器上账户的任何更新都将立即应用到所有的iBMC。
4. 高效性：可以将所有iBMC智能管理系统的用户管理，权限分配，有效期管理都集中到目录服务器上，避免大量的重复性用户配置任务，提高管理效率。
5. 支持性：支持Active Directory和openldap，支持NTLM认证机制。

iBMC LDAP特点：

- 从安全考虑，iBMC只支持LDAPS，支持NTLM鉴权机制。
- 为了确保LDAP服务器的真实性，LDAP支持对服务器合法性验证功能，该功能开启后必须将LDAP服务器的根CA证书导入到iBMC才能使用LDAP功能，且域控制器地址必须配置为与根CA证书里的证书使用者通用名称一致，因为在验证服务器合法性时会匹配域控制器地址与根CA证书的使用者名称是否完全一致。

- 支持多域功能，可配置最多6个域服务器，可指定登录到哪个域或自动匹配域。
- 支持在登录Web或通过SSH方式登录CLI时，使用LDAP账号。
- 支持微软AD和OpenLDAP的LDAP服务端。

**图 2-31** 目录服务原理



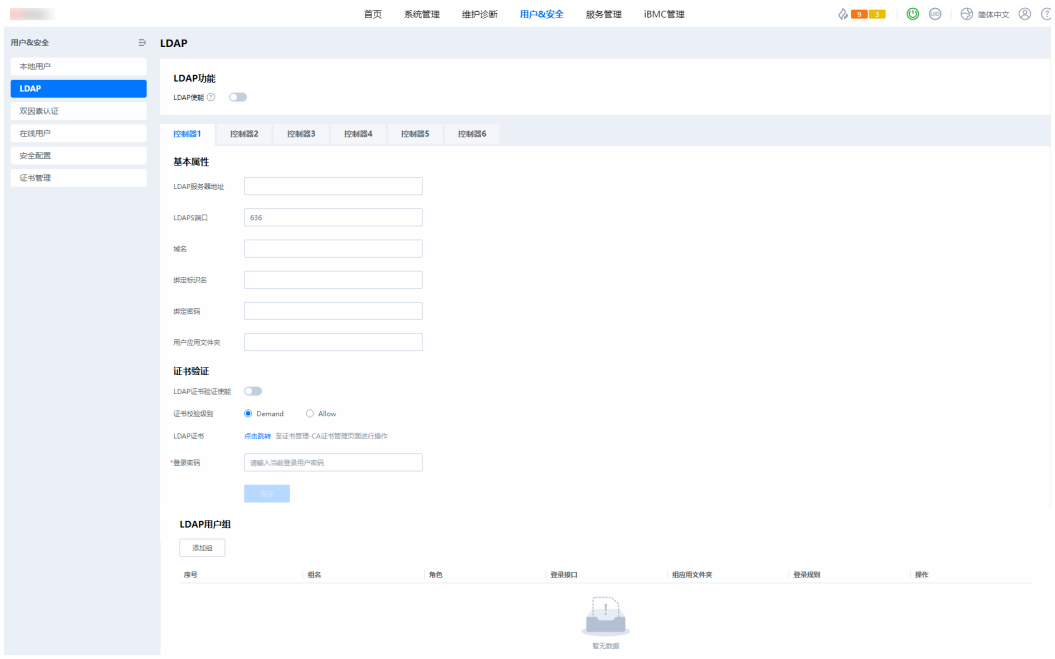

打开“LDAP用户”界面，如图2-32所示。

**说明**

LDAP（Lightweight Directory Access Protocol）是一个访问在线目录服务的协议。LDAP目录中可以存储例如电子邮件地址、邮件路由信息等各种类型的数据，为用户提供更集中、更便捷的查询。

在图2-32中，可以显示或配置LDAP用户的相关信息。

**图 2-32** LDAP 用户界面




通过LDAP用户界面可以完成的设置有：

- 启动或者禁止LDAP
- 启用证书验证
- 设置LDAPS的端口号，默认为636
- LDAP服务器CA根证书导入
- 设置域控制器地址

  域控制器地址为活动目录active directory所在服务器的IP地址或域名。域控制器地址最大长度为255个字符。
- 设置用户角色组名

  组名为配置活动目录active directory中登录iBMC Web界面的用户角色组的名称。组名最大长度为32个字符。
- 设置用户角色组域

  组域为配置活动目录active directory中登录iBMC Web界面的用户角色组的域。组域最大长度为255个字符。
- 设置用户角色组特权

  组特权为配置活动目录active directory中登录iBMC Web界面的用户角色组的特权。包括：规则1、规则2、规则3、web、ssh、redfish权限。

# 2.9 安全管理

## 2.9.1 证书管理

证书是指SSL证书，在建立Web HTTPS连接时使用，用于证明Web站点的身份。

证书管理就是指对SSL证书的各种管理操作，包括查看当前证书信息（证书的使用者、颁发者、有效期、序列号）、生成CSR文件、导入由CSR生成的签名证书（只有公钥，

PEM或者DER格式）、导入自定义证书（包含公钥和私钥，pkcs#12格式，支持对私钥设置密码），同时支持证书过期前告警提示。

openUBMC的SSL证书默认使用自签名SSL证书，证书的签名算法使用RSA-PSS，哈希算法使用SHA512，公钥长度为4096bit，从安全考虑，建议客户在首次使用时导入自己的证书来替换系统中默认的自定义证书，openUBMC提供了两种替换自签名证书的方法：

**第一种方法（使用openUBMC生成的证书）：**

1. 登录到openUBMC Web，修改证书使用者信息；
2. 生成CSR；
3. 导出CSR；
4. 将CSR提交给CA机构；
5. CA机构生成签名证书；
6. 将签名证书导入到openUBMC；
7. 重启openUBMC生效。

**说明**

签名证书必须与CSR配套，即：签名证书必须是通过该CSR申请的，否则导入证书失败。

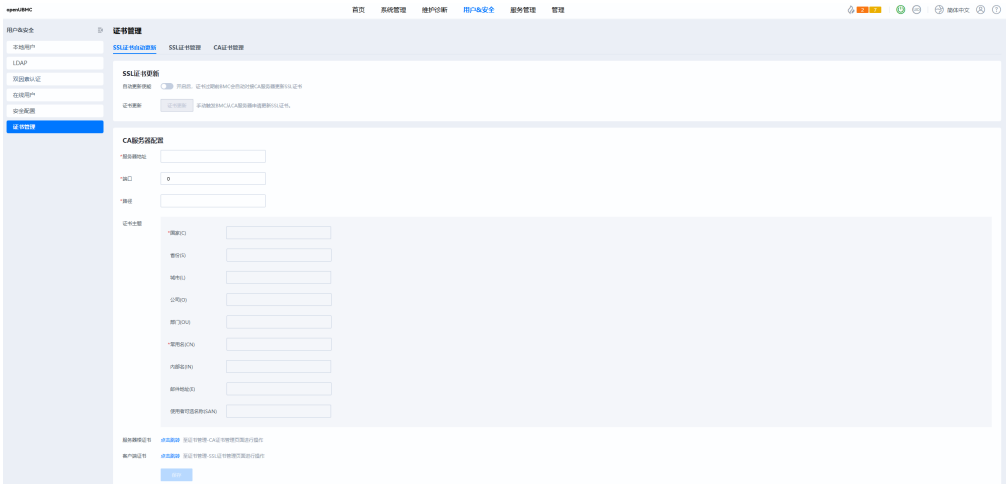
**第二种方法（使用用户提供的证书）：**

1. 用户生成自定义证书或直接从CA购买证书；
2. 登录到openUBMC Web，将自定义证书或购买的证书导入到openUBMC；
3. 重启openUBMC生效。

**图 2-33** SSL 证书管理界面


**第三种方法（SSL证书自动更新）：**

1. 登录到openUBMC Web；
2. 进入“SSL证书自动更新”界面；



3. 根据客户提供的CA服务器的相关配置信息及与CA交互的证书信息，进行对接CA的相关配置
4. 可以通过界面手动触发证书更新，或者通过使能自动更新功能。如果触发了自动更新功能，BMC后台会在证书过期告警前一天自动启动对接CA更新证书。

## 2.9.2 安全协议

外部接入访问默认使用SFTP、SSH、HTTPS、SNMPv3、RMCP+(IPMILAN)方式，传输通道通过使用安全协议进行加密。不安全协议HTTP、SNMP v1/v2c RMCP(IPMILAN)都默认关闭。

各种安全传输协议的特性如下：

SSH：

1. 支持用户密码认证和公钥认证。
2. 支持SSH V2。
3. 支持安全的加密算法aes128-ctr、aes192-ctr、aes256-ctr、aes128-gcm、aes256-gcm、chacha20-poly1305。

SFTP：

1. 仅/tmp目录具有上传、下载文件的权限。
2. 上传到/tmp目录的文件默认不具备可执行权限。

HTTPS：

1. 支持TLS1.2及以上版本。为保持浏览器兼容性，默认开启TLS1.2/TLS1.3，用户可以登录openUBMC禁用TLS1.2。

SNMPv3：

1. 认证算法支持MD5、SHA、SHA256、SHA384、SHA512，支持用户配置，为了满足安全要求，默认配置为SHA256。
2. 加密算法支持DES、AES、AES256，支持用户配置，为了满足安全要求，默认配置为AES。

## 2.9.3 数据保护

openUBMC上涉及密码、密钥的所有敏感数据都进行了加密保护，防止敏感信息泄露。

openUBMC支持升级包的加密和签名保护，防止升级包内容被破解和篡改，保证升级包的机密性和完整性。

除了加密保护，openUBMC对linux shell进行了封装，用户通过SSH、串口等接口登录后无法直接访问文件系统中的文件，防止文件被破坏及软件信息泄露。

openUBMC中支持对关键数据文件进行备份及计算并保存文件校验和，并提供了文件校验失败的备份恢复机制，防止因系统异常掉电导致的数据文件破坏，保护数据文件的可用性和完整性。

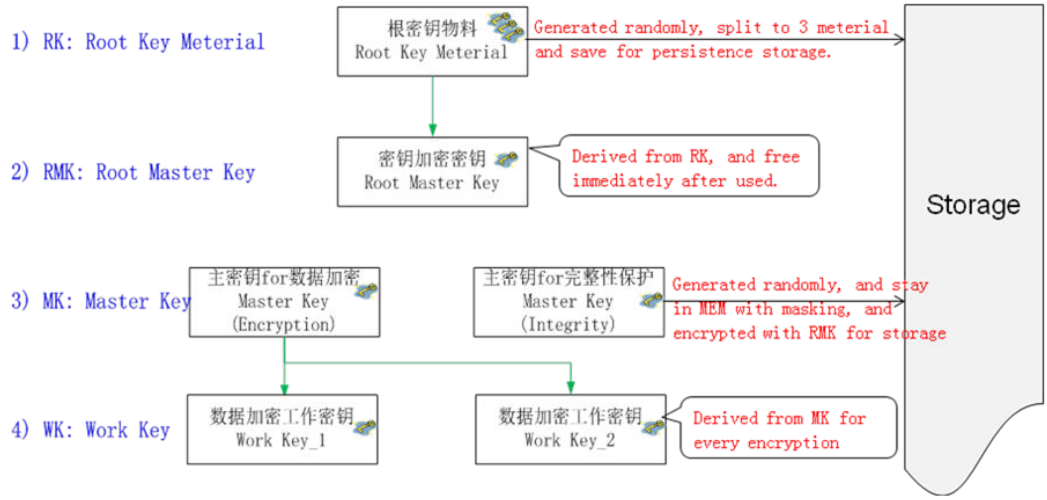
**表 2-5** openUBMC 数据加密情况

| 数据 | 加密算法 |
| --- | --- |
| BMC用户密码（Web/Redfish/SSH/SFTP/串口认证） | SHA512 |
| TLS传输 | ECDHE-RSA-AES256-GCM-SHA384<br>ECDHE-ECDSA-AES256-GCM-SHA384<br>DHE-RSA-AES256-GCM-SHA384<br>ECDHE-RSA-AES128-GCM-SHA256<br>ECDHE-ECDSA-AES128-GCM-SHA256<br>DHE-RSA-AES128-GCM-SHA256<br>ECDHE-RSA-CHACHA20-POLY1305<br>ECDHE-ECDSA-CHACHA20-POLY1305<br>TLS_CHACHA20_POLY1305_SHA256<br>TLS_AES_256_GCM_SHA384<br>TLS_AES_128_GCM_SHA256 |
| SNMP V3加密密码 | DES、AES、AES256 |
| SNMP V1/V2C团体名 | AES256 |
| RMCP+用户密码 | AES256 |
| SSL证书 | AES256 |
| 升级包 | AES256 |
| LDAP域控制器绑定密码 | AES256 |
| VNC密码 | AES256 |

除了对保存在openUBMC中的敏感数据进行加密保护，系统运行过程中产生的敏感数据在使用完后会使用清空内存的方式立刻清空。

## 2.9.4 秘钥管理

openUBMC密钥管理采用分层管理、安全隔离的设计思路，支持多层次密钥架构，上层密钥只用来保护下层密钥，最下层密钥(WK: Work Key)用来保护业务模块应用的机密数据。密钥管理如下图所示。




- 密钥生成：根密钥由安全随机数生成，分成多个组件分开保存；工作密钥使用安全随机数派生而来。
- 密钥使用：密钥用途单一，每个密钥只用于一种用途。
- 密钥存储：根密钥在纯软件环境下由硬编码和文件中两部分存储，基于硬件保护根密钥时根密钥直接存储于硬件中。
- 密钥更新：支持手动更新，执行更新密钥的命令，系统会随机生产新的密钥，旧密钥会被销毁。

## 2.9.5 系统加固

系统最小化安装，openUBMC中对嵌入式linux系统进行裁剪，只安装系统必须的组件，不使用的组件和命令都被删除。

对linux shell命令行进行了封装加固，屏蔽了对linux系统命令的支持，只能执行白名单定义的命令，降低攻击风险。

对系统中SSH、Nginx等服务端进行安全配置加固，只支持安全的算法，不安全的协议和端口默认关闭。

**安全切面**：

openUBMC支持使用安全切面，针对不同产品诉求，对组件运行阶段可以执行的命令类操作进行白名单配置。在组件被外部攻击后，防护组件无法执行恶意的shell命令，来防护系统安全。

当前支持对如下函数进行防护：

popen、system、execl函数簇（execl，execlp，execle，execv，execvp，execvpe，execve）

## 2.9.6 日志审计

openUBMC支持日志审计，日志信息中包含用户名、用户IP地址、操作时间、操作内容等信息。openUBMC会记录SEL日志、操作日志、运行日志、安全日志，并可以通过openUBMC提供的接口进行查阅和审计。

openUBMC日志实时保存在openUBMC的Flash文件系统中，当日志快达到最大存储容量时会产生日志快满的日志提醒，当日志文件达到指定大小后会自动进行日志文件备份。按照最小权限原则，非授权用户无法查看和下载日志文件。

openUBMC支持日志的syslog远程转储，把日志存储到远程syslog服务器中，防止本地日志满后被覆盖丢失，支持对syslog服务器进行验证。

# 2.10 管理接入

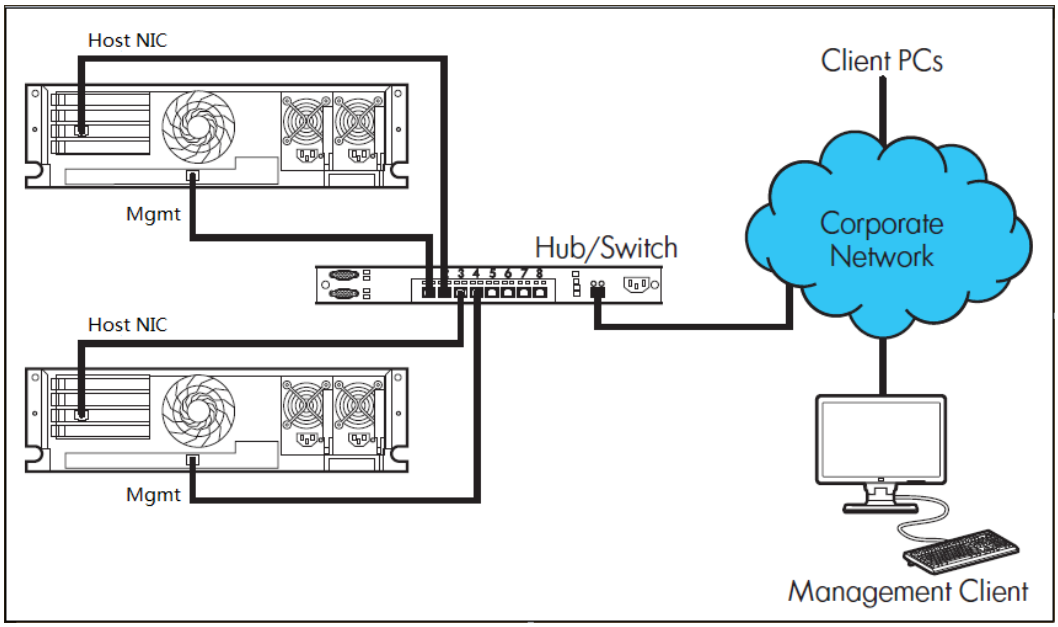
openUBMC兼容支持了IPv4和IPv6两种协议版本地址，支持通过专用管理网口或共享网口(利用NCSI边带功能)接入，专用管理网口和共享网口均支持VLAN功能。

## 2.10.1 管理网口自适应

机架和节点服务器有两个物理管理网口：一个千兆专用管理网口和一个边带管理网口(NCSI，与主机系统共用物理网口)，此功能是根据网口link状态，自动将逻辑网口与其中一个物理网口适配。

网口自适应启用后，服务器更换组网后只要专用管理网口或边带管理网口任一网口连接了网线即可访问管理界面，平滑切换，不需要再配置任何网络信息，省去繁杂的配置步骤，提升维护效率。

**图 2-34** 管理组网图




网口自适应配置界面提供了网口模式查询和设置接口，若选择自适应模式，则可指定某个主机网口作为边带网口，默认为网口1，如图2-35所示。

**图 2-35** 网口自适应配置界面

## 2.10.2 边带管理

边带管理(openUBMC界面称共享网口)就是利用边带(NC-SI)技术使管理系统与主机系统共用主机物理网口，通过一个网口就可以同时做管理操作和业务处理，简化组网，节省交换机端口；从业务数据优先角度考虑，管理数据最大带宽限制在100Mb/S；从安全考虑，利用VLAN技术将管理与业务划分在不同网段。

**图 2-36** 边带管理框图

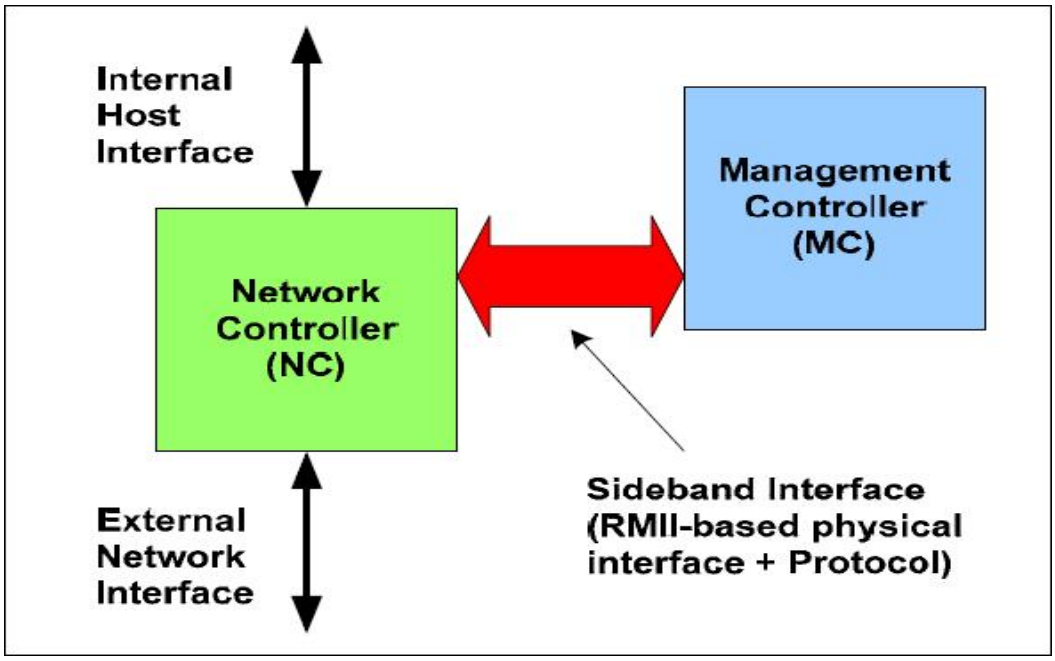
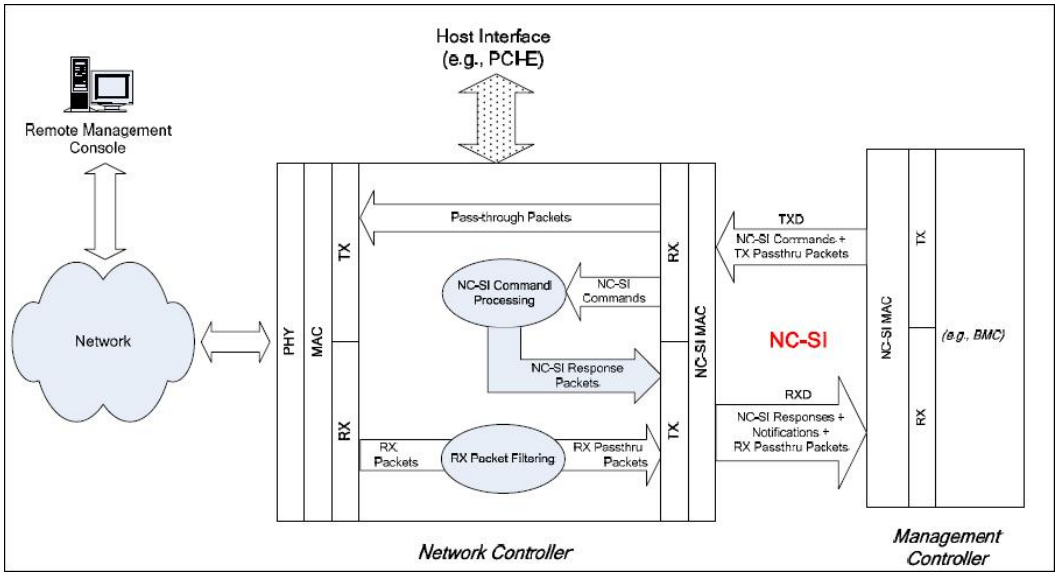



**图 2-37** 边带管理数据流图




## 2.10.3 IPv6

IPv4地址资源很快面临枯竭，解决办法是使用IPv6地址，openUBMC已经正式全面支持了IPv6地址功能。目前openUBMC的WEB、SSH、SNMP、IPMI LAN、redfish接口

模块都已支持IPv6地址访问，专用管理网口和共享网口(NCSI)的物理通道也都支持IPv6地址访问。

**图 2-38** IPv6 地址配置界面

支持手动设置或DHCPv6获取openUBMC的IPv6地址。

# 2.11 配置管理

## 2.11.1 配置导入导出

配置导入导出，就是指把BMC、BIOS所有配置能以配置文件的方式导出和导入，其中RAID控制器配置需在系统POST完成之后导出才有效。此功能提供了一种方法让客户可以轻松的远程保存服务器配置，一旦设备需要更换，可以导入以前保存的配置到新机器，快速完成新设备的配置，也可以针对同一类型机器，用同一个配置文件进行批量配置导入，完成大规模设备的配置和部署。当前支持的接口有：SNMP、CLI、WEB和redfish。

openUBMC也支持通过配置包（zip格式）方式实现多样化配置导入导出能力，从而满足更多场景的配置更新诉求，当前配置包支持的导入能力包括：WEB图片偏好定制、简单固件包升级（仅支持不引起BMC复位的固件包）、配置文件更新、设置还原点。在WEB接口操作界面执行导入导出操作时，支持选择配置文件（JSON格式）和配置包（zip格式）两种方式。

WEB接口操作界面如下图。

**图 2-39** WEB 接口操作界面

**图 2-40** 导出配置文件或配置包操作界面


# 2.12 存储管理

## 2.12.1 RAID 与硬盘管理

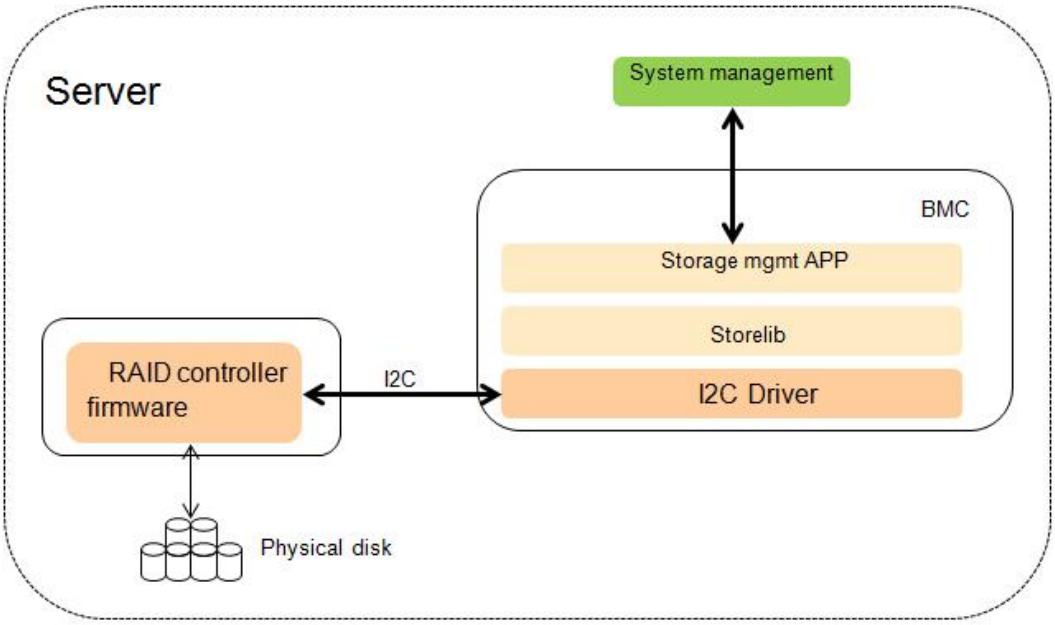
硬盘在服务器中扮演着非常重要的角色，上面安装了系统OS或存储了用户数据，因此对硬盘的管理和监控是非常必要的。

openUBMC通过与RAID控制器交互来对硬盘进行带外管理，依赖于RAID控制器firmware的能力，目前只有最新硬件版本的MegaRAID 9560-16i/MegaRAID 9560-8i/SP686C-M-16i系列等RAID卡支持，如**图 1 硬盘带外管理原理**。注意：产品中插多RAID卡场景下，若其中有RAID卡不支持带外管理技术，则该产品无法支持硬盘带外管理。

**说明**

RAID卡相关功能需要集成厂商的SDK库

**图 2-41** 硬盘带外管理原理




硬盘带外管理包括对RAID控制器、物理盘和逻辑盘管理，支持的特性如表2-6、表2-7、表2-8、表2-9：

**表 2-6** 硬盘带外管理支持属性（状态监控和信息查询）

| 部件 | 管理属性 | 备注 |
| --- | --- | --- |
| RAID控制器 | 名称、序列号、类型、健康状态、固件版本、是否支持带外管理、支持的RAID级别、工作模式、设备接口、SAS地址、支持的条带大小范围、高速缓存存储器大小、SAS速率、是否保留高速缓存、是否打开物理盘故障记忆、SMART错误时回拷、JBOD模式、一致性校验功能、启动设备、PCIe带宽、DDR可纠正ECC计数、PHY误码计数、驱动名称、驱动版本、BBU名称、BBU在位状态、BBU健康状态 | 支持Web/SNMP/CLI/Redfish接口和一键收集；DDR可纠正ECC计数、PHY误码计数不在Web显示。 |
| 物理盘 | 健康状态、序列号、型号、容量、固件版本、介质类型、总线协议、是否热备盘、厂商、重构进度、是否在巡检、medium error计数、prefail计数、其它错误计数、支持速率、协商速率、SAS地址、逻辑归属位置、电源状态、温度、SSD盘剩余寿命、SMART预告警状态、剩余磨损率、定位状态、接口类型 | 支持Web/SNMP/CLI/Redfish接口和一键收集；medium error计数、prefail计数、其它错误计数不在Web显示。 |

| 部件 | 管理属性 | 备注 |
| --- | --- | --- |
| 逻辑盘 | 名称、运行状态、RAID级别、读策略、写策略（默认的和当前的）、条带大小、容量、物理盘写cache是否使能、是否在进行数据一致性校验、成员盘列表、span depth、Number of Drives Per Span、系统盘符、启动设备 | 支持Web/SNMP/CLI/Redfish接口和一键收集 |
| 日志 | RAID卡日志明文导出 | 包含在一键收集中，自动将二进制转为明文收集，不依赖工具解析 |

**说明**

驱动名称、驱动版本、系统盘符这些信息只有安装了对应的带内软件才支持。

**表 2-7** 配置功能点（仅 RAID 卡支持带外管理时支持）

| 部件类型 | 功能点 |
| --- | --- |
| RAID控制器 | Copyback设置、SMART错误时回拷设置、JBOD模式设置、重置控制器 |
| 物理盘 | 全局局部热备状态设置、固件状态设置、物理盘定位设置 |
| 逻辑盘 | 支持逻辑盘的创建、删除和属性修改，可以修改的属性有：VD名称修改、读策略修改、写策略修改、IO策略修改、访问策略修改、后台初始化使能设置、SSD Caching使能设置，CacheCade逻辑盘设置、Disk Cache Policy设置、启动盘设置 |

**表 2-8** 故障监控点

| 部件 | 故障类型及场景 |
| --- | --- |
| RAID控制器 | 内部故障、内存UCE计数非0、内存ECC计数超门限、NVRAM错误计数非0、BMC访问失败 |
| 物理盘 | 故障、预故障(predictive failed error为非0)、重构失败、盘在位但RAID卡不能识别 |
| 逻辑盘 | 逻辑盘状态为offline则该逻辑盘下不在位的物理盘报“In Critical Array”、逻辑盘状态为degraded或partial degraded则该逻辑盘下不在位的物理盘报“In Failed Array” |
| BBU | 电压低、BBU故障、不在位 |

**表 2-9** NVMe 盘管理

| 项目 | 具体取值 |
| --- | --- |
| 信息查询 | 序列号、型号、接口类型、厂商、固件版本、剩余寿命百分比、基于BMA 2.0获取：接口最大速率、接口协商速率、接口类型、介质类型、容量、累计通电时间 |
| 故障监控 | 故障、过温、剩余寿命不足 |

硬盘带外管理界面视图，是基于存储部件逻辑关系组织的。

**图 2-42** RAID 控制器管理界面

**图 2-43** 逻辑盘管理界面

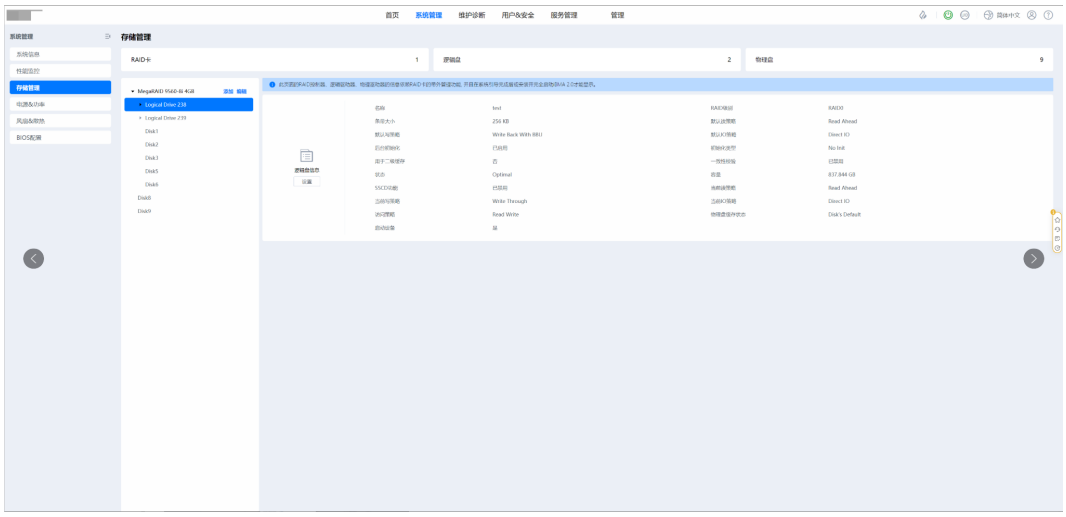


**图 2-44** 物理盘管理界面

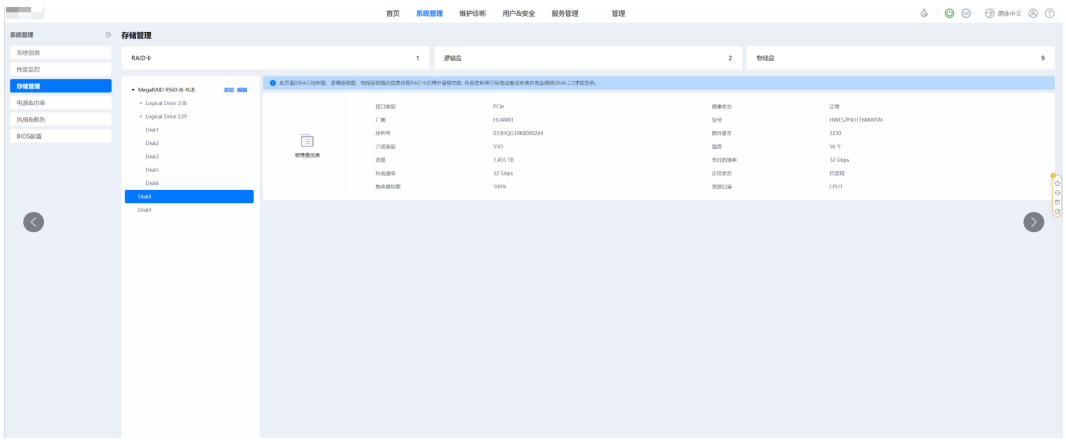


**图 2-45** RAID 配置界面

## 2.12.2 RAID 带外升级

当前RAID卡管理的硬盘固件升级通常依赖OS侧工具，需要重启系统，对业务产生影响、运维成本高、效率低。新支持通过BMC对RAID卡带外热升级功能(**需要支持该能力的RAID卡**)，升级不影响业务。

**图 2-46** 可升级 RAID 卡展示界面

## 2.12.3 硬盘带外升级

当前RAID卡管理的硬盘固件升级通常依赖OS侧工具，需要重启系统，对业务产生影响、运维成本高、效率低。新支持通过BMC带外通道提供RAID卡(**需要支持该能力的RAID卡**)管理SAS/SATA硬盘的固件升级能力，使运维人员可通过北向接口完成升级、查询进度/结果，并与固件管理组件的统一升级框架对齐，提升固件治理的一致性与可维护性。

**图 2-47** 可升级硬盘展示界面


**图 2-48** 硬盘升级界面

**图 2-49** 硬盘待生效界面


**图 2-50** 硬盘生效启动界面


**图 2-51** 硬盘生效完成界面


## 2.12.4 NVMe 盘带外升级

当前NVMe硬盘的固件升级通常依赖OS侧工具，需要登录客户OS、运维成本高、效率低。新支持通过BMC带外通道提供NVMe硬盘(**需要支持NVMe-MI Over MCTP Over PCIe**)的固件升级能力，使运维人员可通过带外完成升级、查询进度/结果，并与固件管理组件的统一升级框架对齐，提升固件治理的一致性与可维护性。同时，固件版本相同的NVMe硬盘支持并行升级，并行升级数量以CSR配置为准。

**图 2-52** 可升级硬盘展示界面


**图 2-53** 硬盘升级界面


**图 2-54** 硬盘升级成功界面


**图 2-55** 硬盘生效启动界面

**图 2-56** 硬盘生效完成界面


**备注：**BMC启动后不能立即升级，需要等南向部件硬盘对象创建和MCTP通道建立，大概登录web后2分钟后可以升级；

## 2.12.5 硬盘故障检测及自愈

RAID卡在检测到硬盘PSM HUNG导致的IO问题的时候，主动对硬盘进行上下电，现网数据统计可通过该策略恢复的硬盘占比20%，提升Raid卡的可靠性，因此盘故障恢复方案需要BMC提供机制对故障槽位硬盘做做日志收集、上下电自愈恢复操作。

支持对指定的SATA盘进行日志收集及上下电自愈恢复机制，功能如下：

1.获取硬盘的固件状态，以判断硬盘是否被raid卡踢盘；

2.对支持收集日志的特定型号和厂商的硬盘，发送直通命令进行日志收集，不同厂商硬盘的直通命令不同，需要区分；

3.收集上来的日志内容为二进制bin文件，文件名包含SerialNumber信息（event_log_{SN}.gz、mark_bad_information_{SN}.gz、flash_id_{SN}.gz）；

4.对硬盘执行上下电操作进行自愈，然后收集日志；

5.每个硬盘的日志收集限制最多执行2次，自愈操作限制最多执行1次；

6.固件状态检测采用三次防抖机制，连续检测到故障状态3次后才触发处理，防止误判。采用上升沿触发机制，持续异常不重复处理，恢复正常后再次异常才会重新触发；

7.每个硬盘槽位的日志文件按SerialNumber分组，最多保留3组文件，超过时自动删除最旧的组；

**图 2-57** RAID 卡日志收集界面

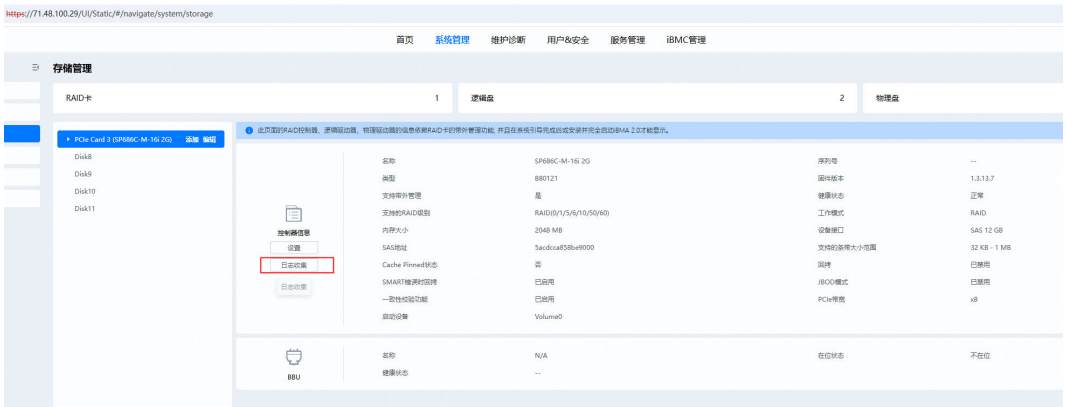
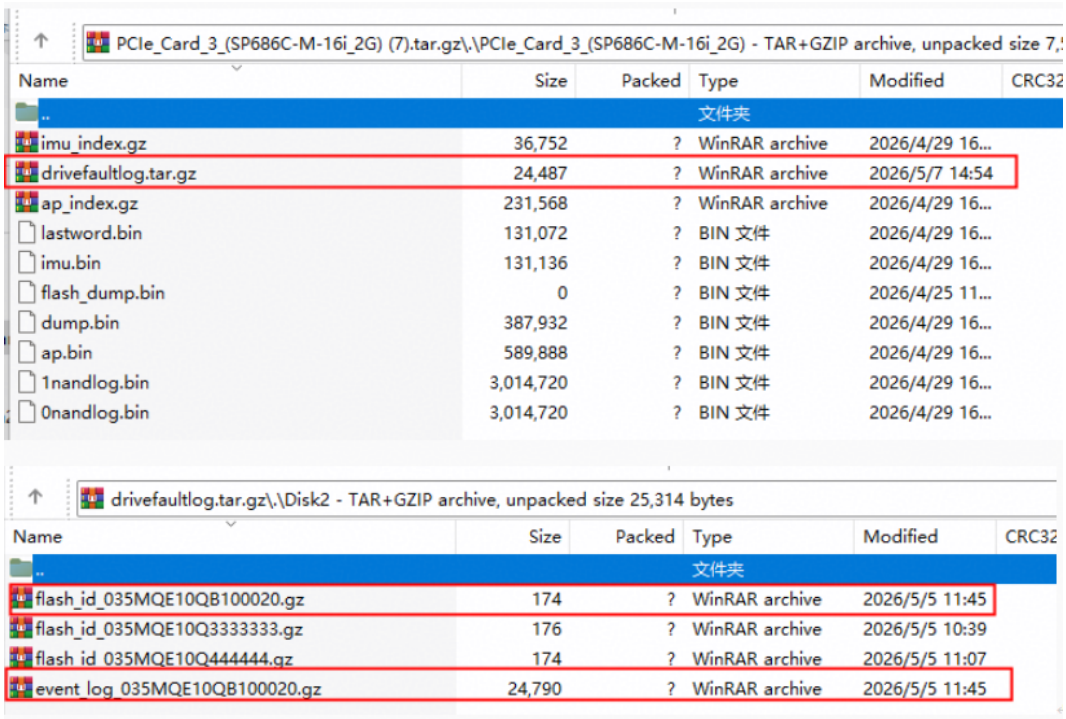


**图 2-58** 硬盘故障后收集到的日志




# 2.13 时间管理

网络时间协议（NTP）：

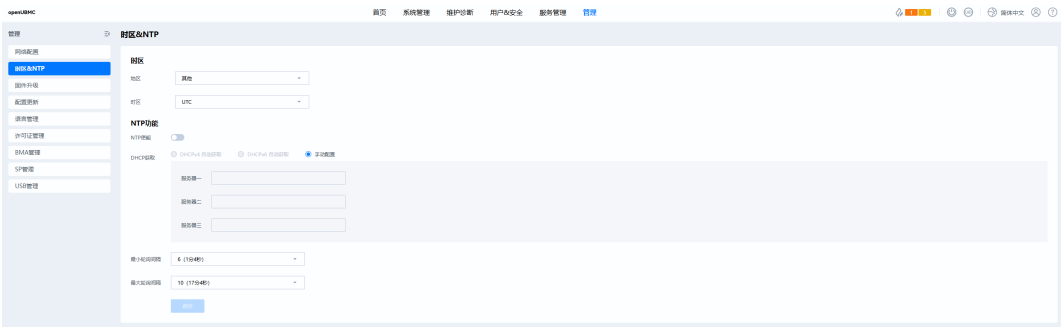
NTP(Network Time Protocol)是用来使计算机时间同步的一种协议。服务器openUBMC自身没有RTC硬件，但支持从多个时间源同步时间且同一时间只能从一个时间源同步，时间源见表2-10。NTP功能默认关闭且支持开启，支持手动设置或自动获取首选和备用NTP服务器地址(支持IP的v4和v6版本)，手动设置时NTP服务器地址还支持FQDN域名输入；从时间获取的安全性考虑，openUBMC支持对NTP服务器合法性校验。

只要NTP功能开启了，无论时间是否同步成功，都不会自动切换到其它时间源。NTP功能关闭，则openUBMC从默认时间源同步时间。时间同步失败、时间跳变都会记录事件日志。

**表 2-10** openUBMC 时间源

| openUBMC | 支持时间源 | 默认时间源 |
| --- | --- | --- |
| 通用服务器 | 主机RTC（BIOS/OS）、NTP | 主机RTC（BIOS/OS） |

**图 2-59** NTP 配置界面



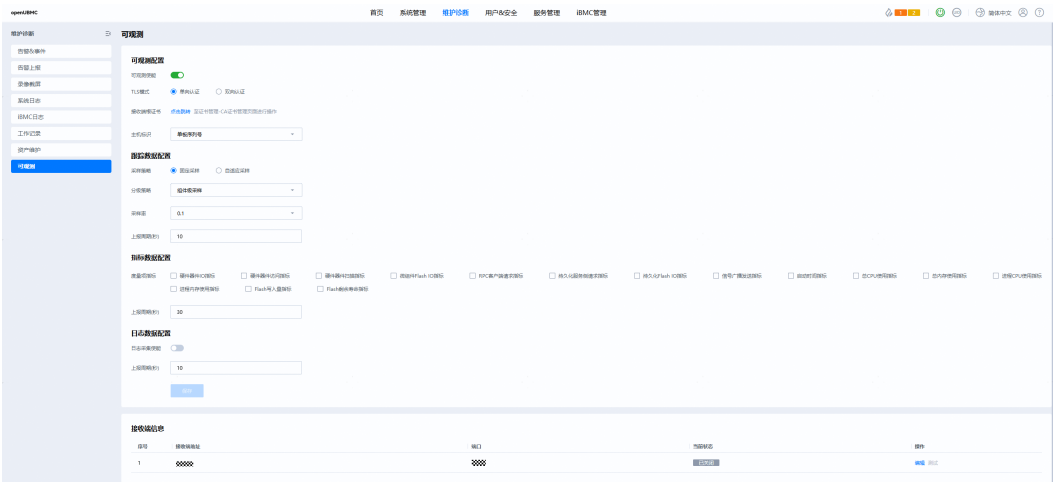
# 2.14 可观测

## 2.14.1 概述

可观测是基于openTelemetry规范构建的可视化能力，主要围绕系统中的可观测数据（包括指标、跟踪和日志）提供采集、上报以及与多种可视化工具集成的能力，帮助用户更好地监控系统性能、预测运行趋势、分析和定位系统故障。

## 2.14.2 可观测配置

**图 2-60** 可观测配置界面



提供可观测配置、跟踪数据配置、指标数据配置、日志数据配置、接收端配置能力。在接收端信息配置界面还提供测试接口，方便用户测试BMC与接收端之间的通道可用状态。（接收通道协议当前只支持OTLP/HTTP协议）。

可观测配置包括：

- 可观测使能
- TLS模式（单向认证、双向认证）
- 接收端根证书、本地证书

跟踪数据配置包括：

- 采样策略（固定采样、自适应采样）
- 分级策略（系统级、组件级、功能级）
- 采样率
- 上报周期

指标数据配置包括：

- 度量项指标（硬件器件IO指标、硬件器件访问器指标、硬件器件扫描指标、微组件flash IO指标、RPC客户端请求指标、持久化服务侧请求指标、持久化flash IO指标、信号广播发送指标、启动时间指标、总CPU使用指标、总内存使用指标、进程CPU使用指标、进程内存使用指标、Flash写入量指标、Flash剩余寿命指标）
- 上报周期

日志数据配置包括：

- 日志使能
- 上报周期

接收端配置包括：

- 接收端地址、端口
- 接收端使能状态

# 2.15 故障管理

## 2.15.1 概述

故障管理是面向服务器提供一系列检测和诊断能力，覆盖光模块和硬盘等多种设备。

## 2.15.2 光模块故障管理

BMC对服务器高速光模块做多种检测和告警：

- 温度异常检测及告警
- 工作电压异常检测及告警
- 接收光功率异常检测及告警
- 发送光功率异常检测及告警
- 发送偏置电流异常检测及告警
- LoS异常检测及告警
- LoL异常检测及告警

BMC对服务器对昇腾NPU参数面光模块异常定位信息做了增强：

- 端口link down记录光模块瞬时定位信息
- 周期记录光模块定位信息

## 2.15.3 硬盘故障管理

BMC支持带外管理硬盘，主要能力如下：

- 对RAID卡管理的HDD，支持获取SMART信息；
- 对RAID卡管理的HDD，支持获取硬盘健康状态；
- 对RAID卡管理的SSD，支持获取SMART信息；
- 对RAID卡管理的SSD及直通NVMe，支持获取获取剩余寿命信息，可通过web界面显示，剩余寿命不足时告警；
- 对RAID卡管理的SSD及直通NVMe，支持获取获取剩可用冗余块信息，并在不足时告警；
- 对RAID卡管理的SSD及直通NVMe，支持记录历史写放大量

## 2.15.4 线缆故障管理

BMC支持在PSR中配置整机UBC线缆白名单，该白名单包含了组件的组合关系和特定接法。通过读取实际接线拓扑与白名单比较，BMC支持对以下场景进行检测：

- 线缆漏插或线缆未插稳
- 线缆错插