

[SSD Test Topic 7] Get disk information on BMC through redfish

1. What is redfish

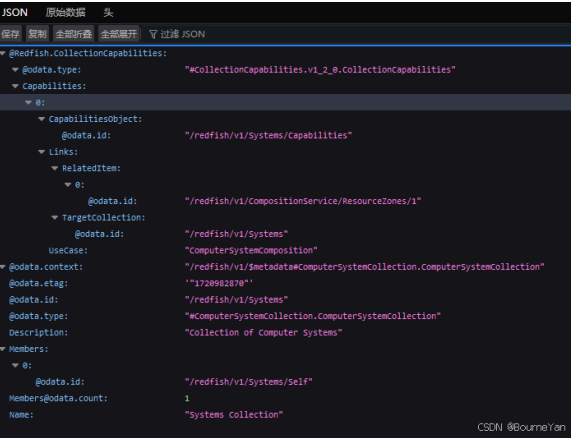
**BMC** Redfish refers to the BMC (Baseboard Management Controller) implementation based on the Redfish API. In computer systems, a BMC is a dedicated microcontroller that is usually integrated on the motherboard of servers, workstations, and other computing devices for remote management and monitoring of these devices.

Redfish API is an open standard developed by DMTF (Distributed Management Task Force) for managing and monitoring hardware devices. BMC Redfish refers to the BMC function implemented through Redfish API, which enables administrators and operation and maintenance personnel to manage servers and related devices in a unified and cross-platform manner through a standardized RESTful interface.

With BMC Redfish, users can perform operations such as obtaining sensor data, configuring device settings, and performing power control through standardized REST APIs without relying on traditional CLI (command line interface) or web-based management interfaces. This standardized interface simplifies device management and automated operations, helping to improve management efficiency and reduce operating costs.

2. How to determine whether the server supports redfish

Normally, when we access the server BMC, we can add /redfish/v1/ after the address to check and confirm. If supported, the browser page will directly return the Json data, as shown in the following figure



3. How to obtain

Due to the support and customization of redfish by various BMC manufacturers, the device information of SSD may be placed in different nodes, some under the drives node, and some under PCIeDevice. Therefore, it can be simply traversed through Python scripts.

Python

```
1 #coding: utf-8
2 from redfish import redfish_client
3 from requests.auth import HTTPBasicAuth
4 import requests
5 # 替换为你的 BMC 的 IP 地址、用户名和密码
6 bmc_ip = '192.168.x.xxx'
7 username = 'xxxx'
8 password = 'xxxx'
9 def get_bmc_management_info(client):
10     """ 获取BMC的管理信息 """
11     try:
12         # 获取BMC的Root资源
13         root_resp = client.get('/redfish/v1')
14
15         #获取Chassis资源子节点上的nvme信息
16         chassis_resp = client.get('/redfish/v1')
17         traverse_ris_object(chassis_resp.obj)
18     except Exception as e:
19         print(f"Failed to retrieve BMC management info: {str(e)}")
20
21 def traverse_ris_object(ris_object, indent=0):
22
```

AI generated projects 登录复制 run

C

AI Assistant

```
twen """ 遍历并打印RisObject对象 """
twen if isinstance(ris_object, dict):
25     for key, value in ris_object.items():
26         if isinstance(value, dict) or isinstance(value, list):
27             print(' ' * indent + f'{key}:')
28             traverse_ris_object(value, indent + 1)
29         else:
30             print(' ' * indent + f'{key}: {value}')
31 elif isinstance(ris_object, list):
32     for item in ris_object:
33         traverse_ris_object(item, indent)
34
35
36 if __name__ == "__main__":
37     # 创建Redfish客户端实例
38     client = redfish_client(bmc_ip, username=username, password=password)
39
40     try:
41         # 连接到BMC
42         client.login()
43
44         # 获取BMC的管理信息
45         get_bmc_management_info(client)
46
47     finally:
48         # 关闭Redfish客户端连接
49         client.logout()
```

You can get the desired information by traversing level by level  
. Take a domestic server as an example. If the information of the SSD device is hung under links->drives,

Python

```
1     for item in chassis_resp.obj['Links']['Drives']:
2         storage_url = item['@odata.id']
3         get_resp = client.get(storage_url)
4         print(get_resp)
5     storage_url = chassis_resp.obj['Links']['Drives']
6     for item in storage_url:
7         disk = client.get(item['@odata.id'])
8         print(disk)
```

It can be obtained, usually including the device's capacity, interface rate, device type, firmware version, temperature, manufacturer information, etc.

4. What else can you do with redfish?

You can also use redfish to perform some BMC operations, such as power on and off, user management, etc. Most of the functions displayed on the BMC page can be operated through the redfish interface

