PowerIT Redfish™ API Guide for Rack PDU and Rack Transfer Switch

Guide Specification

The information contained in this document is subject to change without notice and may not be suitable for all applications. While every precaution has been taken to ensure the accuracy and completeness of this document, Vertiv assumes no responsibility and disclaims all liability for damages result from use of this information or for any errors or omissions.

Refer to local regulations and building codes relating to the application, installation, and operation of this product. The consulting engineer, installer, and/or end user is responsible for compliance with all applicable laws and regulations relation to the application, installation, and operation of this product.

The products covered by this instruction manual are manufactured and/or sold by Vertiv. This document is the property of Vertiv and contains confidential and proprietary information owned by Vertiv. Any copying, use, or disclosure of it without the written permission of Vertiv is strictly prohibited.

Names of companies and products are trademarks or registered trademarks of the respective companies. Any questions regarding usage of trademark names should be directed to the original manufacturer.

Technical Support Site

If you encounter any installation or operational issues with your product, check the pertinent section of this manual to see if the issue can be resolved by following outlined procedures.

Visit https://www.vertiv.com/en-us/support/ for additional assistance.

TABLE OF CONTENTS

1 Introduction

1.1 Redfish™ Logo

1.2 Notice to Users

Vertiv reserves the right to make changes to this document without notice to any user or reseller of this product. Vertiv also reserves the right to substitute or terminate distribution of this document, with no obligation to notify any person or party of such substitutions or terminations.

1.3 Copyrights

1.4 Trademarks

Certain trademarks contained herein are registered to Vertiv Group Corp. Other trademarks may be registered to DMTF (formerly known as the Distributed Management Task Force).

1.5 Use and Disclosure Restrictions

The software and documentation contained in this publication are copyrighted materials.

1.6 Document Usage

All reasonable efforts have been made to provide the accuracy of this document from any technical or typographical errors or missions. Vertiv and its affiliates disclaim responsibility for any labor, materials, or costs incurred as a result of usage of this document. No shall Vertiv and its affiliates be liable for any damages, inclusive of loss of profits or data, arising from the use of or in connection with this document.

1.7 Reporting Document Errors

Should you discover any error or identify a deficiency in this document, please take time to contact us at the following email address: Vertiv-Documents@vertiv.com. Please be sure to provide us with the document name, part number, and page number(s). Also, please provide us with description of the error or the deficiency for the document. If you would like for us to contact you, please provide us with your name and contact information. Thank you for your time. We appreciate any comments and feedback you can provide.

1.8 Document Revision

Revision	Notes
1.0.0	Initial release.

1.9 API Version

Version	Product	Notes
1.0.0	v7.0.2	Initial release.

2 Vertiv™ PowerIT Redfish™ Specification

This PowerIT Redfish™ API specification is intended to inform developers, integrators, and other consumers about the Vertiv PowerIT implementation of the PowerIT Redfish™ standard.

The following PowerIT Redfish™ schemas are supported in some capacity (not exhaustive):

Schema	Version	Schema	Version
AccountService	1.10.0	ManagerAccount	1.4.0
ActionInfo	1.1.2	ManagerNetworkProtocol	1.5.0
Certificate	1.0.0	MessageRegistryFile	1.1.0
CertificateLocations	1.0.0	Outlet	1.4.0
CertificateService	1.0.0	PowerDistribution	1.2.1
Chassis	1.19.0	PowerDistributionMetrics	1.3.0
Circuit	1.5.0	PowerEquipment	1.2.0
ComputerSystem	1.16.0	Role	1.2.2
EthernetInterface	1.4.1	Sensor	1.5.0
EventService	1.5.0	ServiceRoot	1.11.0
JsonSchemaFile	1.0.2	SessionService	1.0.2
LogEntry	1.8.0	TaskService	1.1.4
LogService	1.1.0	TelemetryService	1.2.1
Manager	1.11.0	UpdateService	1.5.0

Since the Vertiv™ PowerIT devices are primarily power equipment, the PowerEquipment schema is particularly useful for device interaction. Another schema that is particularly useful is the Sensor schema, as a multitude of supported sensors will use it to deliver their readings. An OEM schema is implemented that may be used to interact with devices controlled using Vertiv Intelligence Director. See VID Secondary Devices on page 42 for more information.

This implementation of PowerIT Redfish™ is generally read-only. Properties which are defined in a schema or standard as writable may not be writable in this implementation. This does not apply to Actions.

2.1 Quick Links

- AccountService information, see AccountService: /redfish/v1/AccountService on the next page.
- SessionService information, see SessionService: /redfish/v1/SessionService on page 8.
- RackPDU information, see RackPDU Collection: /redfish/v1/PowerEquipment/RackPDUs on page 20.
- TransferSwitch information, see TransferSwitch Collection: /redfish/v1/PowerEquipment/TransferSwitches on page 20.
- Phase information, see Phases on page 24.
- Source information, see Sources on page 26.
- Line information, see Lines on page 26.
- Outlet information, see Outlet Collection: /redfish/v1/PowerEquipment/.../Outlets on page 27.

- Sensor information, see Sensor Collection:/redfish/v1/Chassis/(ChassisID)/Sensors on page 35.
- Event information, see EventService: /redfish/v1/EventService on page 31.

This API is accessible via IPv4 or IPv6 and over HTTP or HTTPS at the ports configured. The lack of a configured IPv4 or IPv6 address, disabled IPv6 connectivity, disabled HTTP, and disabled HTTPS should all impact the Vertiv™ PowerIT Redfish™ interface accordingly. Some standard Redfish™ actions and methods of interacting are unsupported or have limited support. This may include query filtering, applying firmware updates via UpdateService, TelemetryService, and other services. For more information about the Redfish™ standard, please refer to the official material available at https://www.dmtf.org/standards/redfish.

When reading the **value** of a property in this document, plain-text can be generally considered to be a string or a number. There are two special types of values. These values are **null** and plain-text values which are enclosed in **enclosing characters**. These enclosing characters are parentheses (()), less-than and greater-than signs (<>), braces ({}), and brackets ([]). If a value is null (without quotation marks), then that represents the special null value in Redfish. If the value is enclosed in one of the previously mentioned enclosing characters, then the enclosing characters themselves determine whether the value is read-only or read-write and if the value represents a list of literal values or a description. See the **Table 2.1** below for a mapping of enclosure characters to value access (read-only or read-write) and value type (description or list of literal values).

Table 2.1 Parameters for Mapping of Enclosure Characters

Enclosure	Access	Type
(example)	Read-only	Description
<example>	Read-only	List of literal values
{example}	Read-write	Description
[example]	Read-write	List of literal values

2.2 AccountService: /redfish/v1/AccountService

This AccountService implementation provides management of local user accounts through a collection of **ManagerAccount** resources located under the account service. The **ManagerAccount** resources enable users to manage their own account information, while administrators may create, delete, and manage other user accounts. Each user is assigned exactly one role with the **RoleId** property in the **ManagerAccount** resource. The value of the **RoleId** property generally identifies a **Role** resource in the **RoleCollection** resource, where a role defines a set of privileges. Only certain pre-defined roles are available. See Role: /redfish/v1/AccountService/Roles/(ID) on page 6 for more information.

Accounts created via PowerIT Redfish™ are separate from remote authentication accounts and local accounts created via other means. At this time, accounts created via PowerIT Redfish™ are limited to primarily accessing PowerIT Redfish™ URIs only. The following properties within the **AccountService** schema are supported:

Property	Value	Description
@odata.id	/redfish/v1/AccountService	
@odata.type	#AccountService.v1_10_0.AccountService	
AccountLockoutDuration	{Period of time in seconds}	Writable via PATCH.
AccountLockoutThreshold	{Number of allowed failed login attempts}	Writable via PATCH.
Accounts, see ManagerAccount Collection: /redfish/v1/AccountService/Accounts on the facing page.		Object
@odata.id	/redfish/v1/AccountService/Accounts	

Property	Value	Description
ActiveDirectory, see ExternalAccountProvider on page 8.		Object
Description	Account Service	
Id	AccountService	
LDAP, see ExternalAccountProvider on page 8.		Object
MaxPasswordLength	(Maximum password length)	
Name	Account Service	
Oem		Object
OpenBMC		Object
@odata.id	/redfish/v1/AccountService#/Oem/OpenBMC	
@odata.type	#OemAccountService.v1_0_0.AccountService	
AuthMethods		Object
BasicAuth	[true/false]	Writable via PATCH.
Cookie	[true/false]	Writable via PATCH.
SessionToken	[true/false]	Writable via PATCH.
TLS	[true/false]	Writable via PATCH.
XToken	[true/false]	Writable via PATCH.
Roles, see Role: /redfish/v1/AccountService/Roles/{ID} on the next page.		Object
@odata.id	/redfish/v1/AccountService/Roles	
ServiceEnabled	<true/false>	

2.3 ManagerAccount Collection: /redfish/v1/AccountService/Accounts

Existing local user accounts will be listed here. Remote authentication user accounts will not be listed here. The following properties are supported in the response to this URI:

Property	Value	Description
@odata.id	/redfish/v1/AccountService/Accounts	
@odata.type	#ManagerAccountCollection.ManagerAccountCollection	
Description	BMC User Accounts	
Members		Array
@odata.id	/redfish/v1/AccountService/Accounts/{ID}	
Members@odata.count	(Number of entries in Members array)	
Name	Accounts Collection	

See ManagerAccount Creation on the next page, ManagerAccount Deletion on the next page, ManagerAccount: /redfish/v1/AccountService/Accounts/{ID} on page 7, and ManagerAccount Modification on page 8 for information on creating accounts, deleting accounts, viewing account details, and modifying account details.

2.3.1 ManagerAccount Creation

A local account may be created by sending a POST request to **/redfish/v1/AccountService/Accounts** with the following request data:

- **UserName**
- **Password:** May be up to 16 characters in length.
- **RoleId:** See the table of pre-defined roles. See Role: /redfish/v1/AccountService/Roles/(ID) below.
- **Enabled:** May be set to **true** or **false**.

201 Created should be returned upon success with the following response header:

- **Location:** The URI of the created ManagerAccount resource. Use this URI to view account details (via a GET request) or delete the account (via a DELETE request).

2.3.2 ManagerAccount Deletion

To delete an account, send an HTTP DELETE request to the URI corresponding to the account to be deleted.

2.3.3 Role Collection: /redfish/v1/AccountService/Roles

Pre-defined roles are listed here. The following properties are supported in the response to this URI:

Property	Value	Description
@odata.id	/redfish/v1/AccountService/Roles	
@odata.type	#RoleCollection.RoleCollection	
Description	BMC User Roles	
Members		Array
@odata.id	/redfish/v1/AccountService/Roles/(ID)	
Members@odata.count	(Number of entries in Members array)	
Name	Roles Collection	

2.3.4 Role: /redfish/v1/AccountService/Roles/(ID)

For members of the /redfish/v1/AccountService/Roles **Members** array, the Role schema determines the supported properties. The following properties are supported for roles:

Property	Value	Description
@odata.id	/redfish/v1/AccountService/Roles/(RoleID)	
@odata.type	#Role.v1_2_2.Role	
AssignedPrivileges	(Privilege)	Array, see **Table 2.2** on the facing page for potential values.
Description	(RoleID) User Role	
Id	(RoleID)	
IsPredefined	true	

Property	Value	Description
Name	User Role	
OemPrivileges	(Empty Array)	Array
RoleId	(RoleID)	

Currently four standard/pre-defined roles are supported. The standard roles and their assigned privileges are listed in the Table 2.2 below.

Table 2.2 Standard/Pre-defined Roles

Role Id	Assigned Privileges
Administrator	Login, ConfigureManager, ConfigureUsers, ConfigureComponents, ConfigureSelf
Operator	Login, ConfigureComponents, ConfigureSelf
ReadOnly	Login, ConfigureSelf
NoAccess	None

NOTE: The NoAccess role is not displayed in the returned RoleCollection.

2.3.5 ManagerAccount: /redfish/v1/AccountService/Accounts/{ID}

For members of the /redfish/v1/AccountService/Accounts Members array, the ManagerAccount schema determines the supported properties. The following properties are supported for accounts:

Property	Value	Description
@odata.id	/redfish/v1/AccountService/Accounts/{AccountID}	
@odata.type	#ManagerAccount.v1_4_0.ManagerAccount	
AccountTypes	Redfish™	Array
Description	User Account	
Enabled	[true/false]	Writable via PATCH. See below.
Id	(AccountID)	
Links		Object
Role, see Role: /redfish/v1/AccountService/Roles/{ID} on the previous page.		Object
@odata.id	/redfish/v1/AccountService/Roles/{RoleID}	
Locked	[true/false]	Writable via PATCH. See below.
Locked@Redfish.All owableValues	false	Array
Name	User Account	
Password	null	Writable via PATCH. See below.
PasswordChangeReq uired	<true/false>	
RoleId	{RoleID}	Writable via PATCH. See below.
UserName	{AccountID}	Writable via PATCH. See below.

See ManagerAccount Creation on page 6 and ManagerAccount Deletion on page 6 for more information on creating and deleting accounts.

2.3.6 ManagerAccount Modification

The following ManagerAccount properties may be modified on an existing account via a PATCH request to the URI corresponding to the account to be modified:

- UserName
- Password
- RoleId
- Enabled
- Locked

Accounts with the **Administrator** role are able to modify an account's **UserName**, **Password**, **RoleId**, **Enabled**, and **Locked** value. Accounts with the **Operator** or **ReadOnly** role are unable to modify any details of any account except for their own **Password** value. See ManagerAccount Creation on page 6 for information on **Password** and **RoleId** requirements. **Enabled** may be set to **true** or **false**. **Locked** may only be set to **false** via PATCH request.

ExternalAccountProvider

Under the **/redfish/v1/AccountService/** URI, the values of the **ActiveDirectory** and **LDAP** keys are **ExternalAccountProvider** objects. Beware of relying on the contents of these objects in this version of the API.

2.4 SessionService: /redfish/v1/SessionService

Vertiv™ PowerIT Redfish™ Session Service is a service used to manage sessions. This allows a user to exchange a username and password for a session token which may then be used in place of a username and password for susbsequent requests.

Upon a successful session creation, the response from the service will contain the session ID and session token in the **Location** and **X-Auth-Token** response headers respectively. These values need to be saved for future operations with the session.

Reference: https://www.dmtf.org/sites/default/files/Redfish_School-Sessions.pdf

The following properties within the **SessionService** schema are supported:

Property	Value	Description
@odata.id	/redfish/v1/SessionService	
@odata.type	#SessionService.v1_0_2.SessionService	
Description	Session Service	
Id	SessionService	
Name	Session Service	
ServiceEnabled	true	
SessionTimeout	{Session timeout in seconds}	Writable via PATCH. See PATCH SessionTimeout below.
Sessions, see Session Collection: /redfish/v1/SessionService/Sessions below.		Object
@odata.id	/redfish/v1/SessionService/Sessions	

2.4.1 PATCH SessionTimeout

The SessionTimeout value can be changed by sending a PATCH request to **/redfish/v1/SessionService**. This SessionTimeout value represents the number of seconds of inactivity that a session can have before the session service closes the session due to inactivity.

- **Value range:**
  - **Minimum:** 30
  - **Maximum:** 86400

2.4.2 Session Collection: /redfish/v1/SessionService/Sessions

This URI may be used to create new sessions and view existing sessions. The following properties are supported in the response to this URI:

Property	Value	Description
@odata.id	/redfish/v1/SessionService/Sessions	
@odata.type	#SessionCollection.SessionCollection	
Description	Session Collection	
Members		Array
@odata.id	/redfish/v1/SessionService/Sessions/(SessionID)	
Members@odata.count	(Number of entries in Members array)	
Name	Session Collection	

See Session: /redfish/v1/SessionService/Sessions/(ID) on the next page for more information on viewing and deleting sessions.

2.4.3 Create Session

A session is created by sending a POST operation to **/redfish/v1/SessionService/Sessions** with the following request data:

```
{
  "UserName": "{username}",
  "Password": "{password}"
}
```

Upon success, an HTTP response status code of 201 Created, the following response headers, and the details of the created session are returned.

- **Location:** The URI of the created session resource. Use this URI to logout (invalidate the session token) by performing a DELETE operation on the URI.
- **X-Auth-Token:** The session token to be used on all subsequent operations.
  Examples: **X-Auth-Token: abcdefg123456ABCDEFG** or **Authorization: Token abcdefg123456ABCDEFG**.

If session creation failed, 401 Unauthorized would be returned.

2.4.4 Session: /redfish/v1/SessionService/Sessions/(ID)

For members of the /redfish/v1/SessionService/Sessions **Members** array, the **Session** schema determines the supported properties. The following properties are supported for sessions:

Property	Value	Description
@odata.id	/redfish/v1/SessionService/Sessions/(SessionID)	
@odata.type	#Session.v1_3_0.Session	
ClientOriginIPAddress	127.0.0.1	
Description	Manager User Session	
Id	(SessionID)	
Name	User Session	
UserName	(Username)	

2.4.5 Delete Session

A session can be terminated by sending a DELETE request to the session URI. If the operation is successful, an HTTP response status code of 200 OK will be returned.

2.5 Manager Collection: /redfish/v1/Managers

In Vertiv™ PowerIT Redfish™, a manager is a systems management entity that can implement or provide access to a PowerIT Redfish™ service. Examples of managers are BMCs (baseboard management controllers), enclosure managers, management controllers, and other subsystems that are assigned manageability functions. An implementation can have multiple managers, which might be directly accessible through a Redfish-defined interface.

- **Services:**
  - **EthernetInterfaces:** A collection of NICs that this manager uses for network communication.
  - **NetworkProtocol:** The network services and their settings that the manager controls.
  - **Reset:** The reset action resets/reboots the manager.

The following properties are supported in the response to this URI:

Property	Value	Description
@odata.id	/redfish/v1/Managers	
@odata.type	#ManagerCollection.ManagerCollection	
Members		Array
@odata.id	/redfish/v1/Managers/(ID)	
Members@odata.count	(Number of entries in Members array)	
Name	Manager Collection	

2.5.1 Manager: /redfish/v1/Managers/bmc

For members of the /redfish/v1/Managers **Members** array, the Manager schema determines the supported properties. The following properties are supported:

Property	Value	Description
@odata.id	/redfish/v1/Managers/bmc	
@odata.type	#Manager.v1_11_0.Manager	
Actions		Object
#Manager.Reset		Object
@Redfish.ActionInfo	/redfish/v1/Managers/bmc/ResetActionInfo	
target	/redfish/v1/Managers/bmc/Actions/Manager.Reset	See POST Manager.Reset Action on the next page for more information.
DateTime	{Current date and time}	Writable via PATCH. See PATCH DateTime on the next page.
DateTimeLocalOffset	(Time offset from UTC)	
Description	Baseboard Management Controller	
EthernetInterfaces, see EthernetInterface: /redfish/v1/Managers/bmc/EthernetInterfaces/(Intf) on page 13.		Object
@odata.id	/redfish/v1/Managers/bmc/EthernetInterfaces	
FirmwareVersion	(Firmware version)	
Id	bmc	
Links		Object. Writable via PATCH.
ManagerForServers		Array
@odata.id	/redfish/v1/Systems/system	
ManagerForServers@odata.count	(Number of entries in ManagerForServers array)	
LogServices		Object
@odata.id	/redfish/v1/Managers/bmc/LogServices	Not supported
ManagerType	BMC	
Model	(Model information)	
Name	OpenBmc Manager	
NetworkProtocol, see ManagerNetworkProtocol: /redfish/v1/Managers/bmc/NetworkProtocol on page 15.		Object
@odata.id	/redfish/v1/Managers/bmc/NetworkProtocol	
Oem		Object
@odata.id	/redfish/v1/Managers/bmc#/Oem	
@odata.type	#OemManager.Oem	
OpenBmc		Object

Property	Value	Description
@odata.id	/redfish/v1/Managers/bmc#/Oem/OpenBmc	
@odata.type	#OemManager.OpenBmc	
Certificates		Object
@odata.id	/redfish/v1/Managers/bmc/Truststore/Certificates	
PowerState	(Power state)	
SerialConsole		Object
ConnectTypesSupported	SSH	Array
MaxConcurrentSessions	15	
ServiceEnabled	true	This does not represent the enabled state of the serial or SSH service.
ServiceEntryPointUUID	(UUID of the Redfish service hosted by the manager)	
Status		Object
Health	<OK/Warning/Critical>	
HealthRollup	<OK/Warning/Critical>	
State	(State of the resource)	
UUID	(UUID for this manager)	

2.5.2 PATCH DateTime

- Condition: Setting DateTime property needs NTP service disabled.
- Request data

```
{
  "DateTime": "2025-07-07T08:50:17+03:00"
}
```

- Response:
  - NTP disabled: 200 OK should be returned
  - NTP enabled: 500 Internal Server Error will be returned.

2.5.3 POST Manager.Reset Action

- Request:
  - URL: /redfish/v1/Managers/bmc/Actions/Manager.Reset
  - ResetType: GracefulRestart, ForceRestart
  - Payload:

```
{
  "ResetType": "ForceRestart"
}
```

- **Response:** 200 OK should be returned and system restarts

2.5.4 EthernetInterfaceCollection: /redfish/v1/Managers/bmc/EthernetInterfaces

The following properties are supported in the response to this URI:

Property	Value	Description
@odata.id	/redfish/v1/Managers/bmc/EthernetInterfaces	
@odata.type	#EthernetInterfaceCollection.EthernetInterfaceCollection	
Description	Collection of EthernetInterfaces for this Manager	
Members		Array
@odata.id	/redfish/v1/Managers/bmc/EthernetInterfaces/(Intf)	
Members@odata.count	(Number of entries in Members array)	
Name	Ethernet Network Interface Collection	

See EthernetInterface: /redfish/v1/Managers/bmc/EthernetInterfaces/(Intf) below for more information

2.5.5 EthernetInterface: /redfish/v1/Managers/bmc/EthernetInterfaces/(Intf)

For members of the /redfish/v1/Managers/bmc/EthernetInterfaces **Members** array, the **EthernetInterface** schema determines the supported properties. The following properties are supported:

Property	Value	Description
@odata.id	/redfish/v1/Managers/bmc/EthernetInterfaces/(Intf)	
@odata.type	#EthernetInterface.v1_4_1.EthernetInterface	
DHCPv4		Object
DHCPEnabled	<true/false>	
UseDNSServers	<true/false>	
UseDomainName	<true/false>	
UseNTPServers	<true/false>	
DHCPv6		Object
OperatingMode	"Stateful"	
UseDNSServers	<true/false>	
UseDomainName	<true/false>	
UseNTPServers	<true/false>	
Description	Management Network Interface	
FQDN	(The fully qualified domain name that DNS obtains for this interface)	
HostName	(The host name without any domain information)	
IPv4Addresses		Array

Property	Value	Description
Address	(IP address)	
AddressOrigin	<DHCP/Static>	
Gateway	(Gateway)	
SubnetMask	(Subnet mask)	
IPv4StaticAddresses		Array
Address	(IP address)	
AddressOrigin	<DHCP/Static>	
Gateway	(Gateway)	
SubnetMask	(Subnet mask)	
IPv6AddressPolicyTable		Array
IPv6Addresses		Array
Address	(IP address)	
AddressOrigin	<LinkLocal/DHCPv6>	
AddressState	null	
PrefixLength	(Prefix length)	
IPv6DefaultGateway	(IPv6 default gateway address)	
IPv6StaticAddresses		Array
Address	(IP address)	
AddressOrigin	<LinkLocal/DHCPv6>	
AddressState	null	
PrefixLength	(Prefix length)	
Id	(Intf)	
InterfaceEnabled	<true/false>	
LinkStatus	(Link status)	
MACAddress	(MAC address)	
MTUSize	0	Not functional
Name	Manager Ethernet Interface	
NameServers		Array
SpeedMbps	0	Not functional
StaticNameServers		Array
Status		Object
Health	<OK/Warning/Critical>	
HealthRollup	<OK/Warning/Critical>	
State	(State of the resource)	

2.5.6 ManagerNetworkProtocol: /redfish/v1/Managers/bmc/NetworkProtocol

The network service settings for the manager. The following properties within the ManagerNetworkProtocol schema are supported:

Property	Value	Description
@odata.id	/redfish/v1/Managers/bmc/NetworkProtocol	
@odata.type	#ManagerNetworkProtocol.v1_5_0.ManagerNetworkProtocol	
Description	Manager Network Service	
FQDN	(The fully qualified domain name for the manager)	
HTTP		Object. Writable via PATCH. See PATCH HTTP/HTTPS/SSH on page 18.
Port	{HTTP port}	Writable via PATCH.
ProtocolEnabled	[true/false]	Writable via PATCH.
HTTPS		Object. Writable via PATCH. See PATCH HTTP/HTTPS/SSH on page 18.
Certificates		Object
@odata.id	/redfish/v1/Managers/bmc/NetworkProtocol/HTTPS/Certificates	
Port	{HTTPS port}	Writable via PATCH.
ProtocolEnabled	[true/false]	Writable via PATCH.
HostName	(The host name without any domain information)	
Id	NetworkProtocol	
NTP		Object. Writable via PATCH. See PATCH NTP on page 17.
NTPServers		Array. Contains currently assigned NTP servers. Writable via PATCH.
ProtocolEnabled	[true/false]	Writable via PATCH.
Name	Manager Network Protocol	
SSH		Object. Writable via PATCH. See PATCH HTTP/HTTPS/SSH on page 18.
Port	{SSH port}	Writable via PATCH.
ProtocolEnabled	[true/false]	Writable via PATCH.
Status		Object
Health	<OK/Warning/Critical>	
HealthRollup	<OK/Warning/Critical>	
State	(State of the resource)	

2.5.7 PATCH NTP

- Supported payload:
  - ProtocolEnabled:
    - True: Set time via NTP
    - Talse: Stop setting time via NTP
  - NTPServers: Supports 2 unique NTP servers (IMD5 limitation)
- Request with valid payload:
  - URL: /redfish/v1/Managers/bmc/NetworkProtocol
  - One server payload:

```
{
  "NTP": {
       "NTPServers": ["1.pool.ntp.org"]
   }
}
```

  - Two servers payload with ProtocolEnabled set to true:

```
{
"NTP": {
  "NTPServers": [
     "2.pool.ntp.org",
     "3.pool.ntp.org"
  ],
  "ProtocolEnabled": true
  }
}
```

  - Response: 204 No Content should be returned with no response data
- Request with invalid payload:
  - URL: /redfish/v1/Managers/bmc/NetworkProtocol
  - Zero server payload:

```
{
  "NTP": {
       "NTPServers": []
   }
}
```

  - Three servers payload:

```
{
  "NTP": {
      "NTPServers": [
         "0.pool.ntp.org",
```

```
          "1.pool.ntp.org",
          "2.pool.ntp.org"
        ],
       "ProtocolEnabled": true
   }
}
```

- Response: 400 Bad Request should be returned with a Base.1.11.0.PropertyValueIncorrect error code. An example response is provided below.

```
{
   "error": {
        "code": "Base.1.11.0.PropertyValueIncorrect",
        "message": "The property 'NTPServers/' with the requested value of '0' could not
be written because the value does not meet the constraints of the implementation."
    }
}
```

2.5.8 PATCH HTTP/HTTPS/SSH

- Supported payload:
  - ProtocolEnabled
  - Port
- Request with valid payload:
  - URL: /redfish/v1/Managers/bmc/NetworkProtocol
  - One protocol payload:

```
{
  "HTTP": {
      "Port": 80,
      "ProtocolEnabled": false
  }
}
```

  - Multiple protocols payload:

```
{
  "HTTP": {
     "Port": 80,
     "ProtocolEnabled": false
   },
   "HTTPS": {
   "Port": 888,
   "ProtocolEnabled": true
   },
   "SSH": {
      "Port": 822,
      "ProtocolEnabled": false
   }
```

```
}
```

- **Response:** 204 No Content should be returned with no response data

- Request with invalid value:
  - **URL:** /redfish/v1/Managers/bmc/NetworkProtocol
  - **In-use port payload:**

```
{
   "HTTP": {
      "Port": 888,
      "ProtocolEnabled": true
    },
    "HTTPS": {
       "Port": 888,
       "ProtocolEnabled": true
    },
       "SSH": {
       "Port": 888,
       "ProtocolEnabled": true
    }
}
```

- **Response:** Returns 400 Bad Request with the following response data

```
{
   "error": {
       "@Message.ExtendedInfo": [
          {
            "Message": "The property 'HTTP/Port' with the requested value of '888' could
not be written because the value does not meet the constraints of the implementation.",
            "MessageArgs": [
                "HTTP/Port",
                "888"
            ],
            "MessageId": "Base.1.11.0.PropertyValueIncorrect"
          },
          {
            "Message": "The property 'SSH/Port' with the requested value of '888' could
not be written because the value does not meet the constraints of the implementation.",
            "MessageArgs": [
                "SSH/Port",
                "888"
          ],
             "MessageId": "Base.1.11.0.PropertyValueIncorrect"
       }
    ],
    "code": "Base.1.11.0.GeneralError",
    "message": "A general error has occurred. See Resolution for information on how to
resolve the error."
   }
}
```

2.6 PowerEquipment: /redfish/v1/PowerEquipment

The PowerEquipment schema is where the vast majority of pertinent information about Vertiv™ PowerIT Rack Power Distribution Units (rPDUs) and Vertiv™ PowerIT Rack Transfer Switches (RTSs) may be found. In the Redfish schema and specification, these are known as **RackPDUs** and **TransferSwitches**. The following properties within the PowerEquipment schema are supported:

Property	Value	Description
@odata.id	/redfish/v1/PowerEquipment	
@odata.type	#PowerEquipment.v1_2_0.PowerEquipment	
Id	PowerEquipment	
Name	DCIM Power Equipment	
RackPDUs, see RackPDU Collection: /redfish/v1/PowerEquipment/RackPDUs below.		Object
@odata.id	/redfish/v1/PowerEquipment/RackPDUs	
TransferSwitches, see TransferSwitch Collection: /redfish/v1/PowerEquipment/TransferSwitches below.		Object
@odata.id	/redfish/v1/PowerEquipment/TransferSwitches	
Status		Object
HealthRollup	OK	
State	Enabled	

2.6.1 RackPDU Collection: /redfish/v1/PowerEquipment/RackPDUs

Identified PowerIT Rack Power Distribution Units (RackPDUs) will be listed here. The following properties are supported in the response to this URI:

Property	Value	Description
@odata.id	/redfish/v1/PowerEquipment/RackPDUs	
@odata.type	#PowerDistributionCollection.PowerDistributionCollection	
Members		Array
@odata.id	/redfish/v1/PowerEquipment/RackPDUs/(ID)	
Members@odata.count	(Number of entries in Members array)	
Name	RackPDU Collection	

See RackPDU: /redfish/v1/PowerEquipment/RackPDUs/(ID) on the facing page for more information.

2.6.2 TransferSwitch Collection: /redfish/v1/PowerEquipment/TransferSwitches

Identified PowerIT Rack Transfer Switches (RTSs) will be listed here. The following properties are supported in the response to this URI:

Property	Value	Description
@odata.id	/redfish/v1/PowerEquipment/TransferSwitches	
@odata.type	#PowerDistributionCollection.PowerDistributionCollection	
Members		Array
@odata.id	/redfish/v1/PowerEquipment/TransferSwitches/(ID)	
Members@odata.count	(Number of entries in Members array)	
Name	Transfer Switch Collection	

See TransferSwitch: /redfish/v1/PowerEquipment/TransferSwitches/(ID) on the next page for more information.

2.6.3 RackPDU: /redfish/v1/PowerEquipment/RackPDUs/(ID)

For members of the /redfish/v1/PowerEquipment/RackPDUs **Members** array, the PowerDistribution schema determines the supported properties. The following properties within the PowerDistribution schema are supported for RackPDUs:

Property	Value	Description
@odata.id	/redfish/v1/PowerEquipment/RackPDUs/(ID)	
@odata.type	#PowerDistribution.v1_2_1.PowerDistribution	
Branches, see Branch Collection: /redfish/v1/PowerEquipment/.../Branches on page 29.		Object
@odata.id	/redfish/v1/PowerEquipment/RackPDUs/(ID)/Branches	
EquipmentType	RackPDU	
FirmwareVersion	(IMD firmware version)	
Id	(ID)	
Links		Object
Chassis, see Chassis on page 34.		Array
@odata.id	/redfish/v1/Chassis/(ChassisID)	
Mains, see Mains Collection: /redfish/v1/PowerEquipment/.../Mains on page 24.		Object
@odata.id	/redfish/v1/PowerEquipment/RackPDUs/(ID)/Mains	
Manufacturer	Vertiv	
Metrics, see Metrics: /redfish/v1/PowerEquipment/.../Metrics on page 30.		Object
@odata.id	/redfish/v1/PowerEquipment/RackPDUs/(ID)/Metrics	
Model	(Model)	
Name	(Name)	
Outlets, see Outlet Collection: /redfish/v1/PowerEquipment/.../Outlets on page 27.		Object
@odata.id	/redfish/v1/PowerEquipment/RackPDUs/(ID)/Outlets	
PartNumber	(Part number)	
SerialNumber	(Serial number)	

Property	Value	Description
Status		Object
Health	OK	
State	Enabled	

The following non-standard properties are also supported for RackPDUs:

Property	Value	Description
ResidualCurrentAggregate	(Decimal value)	The aggregate residual current in mA. Only the pre-decimal point portion is significant. Only returned if the device is capable of providing a value.

2.6.4 TransferSwitch: /redfish/v1/PowerEquipment/TransferSwitches/(ID)

For members of the /redfish/v1/PowerEquipment/TransferSwitches **Members** array, the PowerDistribution schema determines the supported properties. The following properties within the PowerDistribution schema are supported for TransferSwitches:

Property	Value	Description
@odata.id	/redfish/v1/PowerEquipment/Transfer Switches/(ID)	
@odata.type	#PowerDistribution.v1_2_1.PowerDistribution	
Branches, see Branch Collection: /redfish/v1/PowerEquipment/.../Branches on page 29.		Object
@odata.id	/redfish/v1/PowerEquipment/Transfer Switches/(ID)/Branches	
EquipmentType	AutomaticTransferSwitch	
FirmwareVersion	(IMD firmware version)	
Id	(ID)	
Links		Object
Chassis, see Chassis Collection: /redfish/v1/Chassis on page 34.		Array
@odata.id	/redfish/v1/Chassis/(ChassisID)	
Mains, see Mains Collection: /redfish/v1/PowerEquipment/.../Mains on page 24.		Object
@odata.id	/redfish/v1/PowerEquipment/Transfer Switches/(ID)/Mains	
MainsRedundancy		Object
MaxSupportedInGroup	2	
MinNeededInGroup	2	
RedundancyGroup		Array
@odata.id	/redfish/v1/PowerEquipment/Transfer Switches/(ID)/Mains/(Source A ID)	

Property	Value	Description
@odata.id	/redfish/v1/PowerEquipment/Transfer Switches/(ID)/Mains/(Source B ID)	
RedundancyType	Failover	
Status		Object
Health	(MainsRedundancy Health)	See MainsRedundancy Health on the next page.
State	Enabled	
Manufacturer	Vertiv	
Metrics, see Metrics: /redfish/v1/PowerEquipment/.../Metrics on page 30.		Object
@odata.id	/redfish/v1/PowerEquipment/Transfer Switches/(ID)/Metrics	
Model	(Model)	
Name	(Name)	
Outlets, see Outlet Collection: /redfish/v1/PowerEquipment/.../Outlets on page 27.	.	Object
@odata.id	/redfish/v1/PowerEquipment/Transfer Switches/(ID)/Outlets	
PartNumber	(Part number)	
SerialNumber	(Serial number)	
Status		Object
Health	(RTS health)	**Critical** if a RTS hardware fault is detected, **OK** otherwise.
State	(RTS state)	**Enabled** if RTS power output is enabled, **Disabled** otherwise.
TransferConfiguration		Object
ActiveMainsId	(ID of active mains circuit)	
AutoTransferEnabled	true	
ClosedTransitionAllowed	false	
PreferredMainsId	(ID of preferred mains circuit)	
RetransferDelaySeconds	(Integer)	
RetransferEnabled	(Boolean)	

The following non-standard properties are also supported for TransferSwitches:

Property	Value	Description
ResidualCurrentAggregate	(Decimal value)	The aggregate residual current in mA. Only the pre-decimal point portion is significant. Only returned if the device is capable of providing a value.

MainsRedundancy Health

The value for the **Health** property in the **MainsRedundancy** object's **Status** object is determined as shown in the table below.

Parameters	Source A Qualified	Source A Unqualified
Source B Qualified	OK	Warning
Source B Unqualified	Warning	Critical

2.6.5 Mains Collection: /redfish/v1/PowerEquipment/.../Mains

Identified Mains circuits (phases, lines, and sources) will be listed here. Supported URIs are of the form /redfish/v1/PowerEquipment/TransferSwitches/(ID)/Mains and /redfish/v1/PowerEquipment/RackPDUs/(ID)/Mains. The following properties are provided in the response to such a URI:

Property	Value	Description
@odata.id	/redfish/v1/PowerEquipment/<TransferSwitches/RackPDUs>/(ID)/Mains	
@odata.type	#CircuitCollection.CircuitCollection	
Members		Array
@odata.id	/redfish/v1/PowerEquipment/<TransferSwitches/RackPDUs>/(ID)/Mains/(MainsID)	
Members@odata.count	(Number of entries in Members array)	
Name	Power Input Circuit Collection	

See Phases below for information on the representation of phases, Lines on page 26 for information on the representation of lines, and Sources on page 26 for information on the representation of sources.

2.6.6 Phases

For members of the **Members** array found at /redfish/v1/PowerEquipment/RackPDUs/(ID)/Mains or /redfish/v1/PowerEquipment/TransferSwitches/(ID)/Mains, the following properties within the Circuit schema are supported for mains circuits which represent phases (such as Phase_A, Phase_AB, etc.):

Property	Value	Description
@odata.id	/redfish/v1/PowerEquipment/<TransferSwitches/RackPDUs >/(ID)/Mains/(MainsID)	
@odata.type	#Circuit.v1_5_0.Circuit	
Actions		Object
#Circuit.ResetMetrics		Object
target	/redfish/v1/PowerEquipment/<TransferSwitches/RackPDUs>/(ID)/Mains (MainsID)/Actions/Circuit.ResetMetrics	
CurrentAmps		Object
CrestFactor	(Decimal value)	
Reading	(Decimal value)	
EnergykWh		Object

Property	Value	Description
Reading	(Decimal value)	
Id	(MainsID)	
Name	Mains (Modified MainsID)	
PolyPhasePower Watts	(Complex object)	Object, see PolyPhasePowerWatts below.
PolyPhaseVoltage	(Complex object)	Object, see PolyPhaseVoltage below.
Status		Object
Health	OK	

PolyPhasePowerWatts

The value of the PolyPhasePowerWatts property is an object. The name or **key** of this object is an ElectricalContext string as defined in the Circuit schema that identifies the electrical context of the contained measurements. Examples include **Line1ToNeutral**, **Line1ToLine2**, and **Line2ToLine3**. The measurements contained within this object correspond to the following properties as found in the **SensorPowerExcerpt** portion of the Circuit schema. The value of the PolyPhasePowerWatts property is detailed below.

Property	Value	Description
(ElectricalContext value)		Object. Property name may be **Line1ToNeutral**, **Line1ToLine2**, etc.
ApparentVA	(Decimal value)	
PowerFactor	(Decimal value)	
Reading	(Decimal value)	

PolyPhaseVoltage

The value of the PolyPhaseVoltage property is an object. The name or **key** of this object is an ElectricalContext string as defined in the Circuit schema that identifies the electrical context of the contained measurements. Examples include **Line1ToNeutral**, **Line1ToLine2**, and **Line2ToLine3**. The measurement contained within this object is the **Reading** property as found in the **SensorVoltageExcerpt** portion of the Circuit schema. The value of the PolyPhaseVoltage property is detailed below.

Property	Value	Description
(ElectricalContext value)		Object. Property name may be **Line1ToNeutral**, **Line1ToLine2**, etc.
Reading	(Decimal value)	

2.6.7 Sources

For members of the /redfish/v1/PowerEquipment/TransferSwitches/(ID)/Mains **Members** array, the Circuit schema determines the supported properties. The following properties within the Circuit schema are supported for mains circuits which represent sources (such as Source_A and Source_B).

Property	Value	Description
@odata.id	/redfish/v1/PowerEquipment/TransferSwitches/(ID)/Mains/(MainsID)	
@odata.type	#Circuit.v1_5_0.Circuit	
CurrentAmps		Object
CrestFactor	(Decimal value)	
Reading	(Decimal value)	
FrequencyHz		Object
Reading	(Decimal value)	
Id	(MainsID)	
Name	Mains (Modified MainsID)	
Status		Object
Health	OK	
State	(Source state)	**Enabled** if source is actively providing power, **StandbySpare** if source is not actively providing power but is qualified, and **Disabled** otherwise.
Voltage		Object
CrestFactor	(Decimal value)	
Reading	(Decimal value)	

2.6.8 Lines

For members of the /redfish/v1/PowerEquipment/RackPDUs/(ID)/Mains **Members** array, the Circuit schema determines the supported properties. The following properties within the Circuit schema are supported for mains circuits which represent lines (such as Neutral_Line, Line_A, Line_B, etc.).

Property	Value	Description
@odata.id	/redfish/v1/PowerEquipment/RackPDUs/(ID)/Mai ns/(MainsID)	
@odata.type	#Circuit.v1_5_0.Circuit	
CurrentAmps		Object. Non-standard implementation. See below for details.
Id	(MainsID)	
Name	Mains (Modified MainsID)	
Status		Object
Health	OK	

The CurrentAmps property is implemented in the following non-standard way for mains circuits which represent lines:

Property	Value	Description
CurrentAmps		Object
(Context)		Object. Property name may be Neutral, Line A, etc.
Reading	(Decimal value)	

2.6.9 Outlet Collection: /redfish/v1/PowerEquipment/.../Outlets

Identified outlets will be listed here. Supported URIs are of the form /redfish/v1/PowerEquipment/TransferSwitches/(ID)/Outlets and /redfish/v1/PowerEquipment/RackPDUs/(ID)/Outlets. The following properties are provided in the response to such a URI:

Property	Value	Description
@odata.id	/redfish/v1/PowerEquipment/<TransferSwitches/RackPDUs>/(ID)/Outlets	
@odata.type	#OutletCollection.OutletCollection	
Members		Array
@odata.id	/redfish/v1/PowerEquipment/<TransferSwitches/RackPDUs>/(ID)/Outlets/(OutletID)	
Members@odata.count	(Number of entries in Members array)	
Name	Outlet Collection	

Outlet

For members of the Members array found at /redfish/v1/PowerEquipment/RackPDUs/(ID)/Outlets or /redfish/v1/PowerEquipment/TransferSwitches/(ID)/Outlets, the following properties within the Outlet schema are supported:

Property	Value	Description
@odata.id	/redfish/v1/PowerEquipment/<TransferSwitches/RackPDUs>/(ID)/Outlets/(OutletID)	
@odata.type	#Outlet.v1_4_0.Outlet	
Actions		Object
#Outlet.PowerControl		Object
target	/redfish/v1/PowerEquipment/<TransferSwitches/RackPDUs>/(ID)/Outlets/(OutletID)/Actions/Outlet.PowerControl	See Outlet PowerControl on the next page for more details.
#Outlet.ResetMetrics		Object
target	/redfish/v1/PowerEquipment/<TransferSwitches/RackPDUs>/(ID)/Outlets/(OutletID)/Actions/Outlet.ResetMetrics	
CurrentAmps		Object
CrestFactor	(Decimal value)	
Reading	(Decimal value)	
EnergykWh		Object
Reading	(Decimal value)	

Property	Value	Description
Id	(OutletID)	
Name	(Modified OutletID)	
PowerEnabled	(Boolean)	**true** if the outlet has measurements (voltage, current, etc.). **false** otherwise.
PowerOffDelaySeconds	(Integer value)	See Outlet PowerControl below for more details.
PowerOnDelaySeconds	(Integer value)	See Outlet PowerControl below for more details.
PowerState	(String)	**On** if the outlet is powered without error. **Off** if the outlet is off, the outlet's parent breaker is open, or if the outlet is in an error state.
PowerWatts		Object
ApparentVA	(Decimal value)	
PowerFactor	(Decimal value)	
Reading	(Decimal value)	
RatedCurrentAmps	(Integer value)	
Voltage		Object
Reading	(Decimal value)	

Outlet PowerControl

The Outlet.PowerControl action is used to start an outlet power transition or cancel an outlet power transition. Triggering this action requires sending a HTTP POST request to the endpoint that contains a body with two key-value pairs. The two required keys are **PowerState** and **Delay**.

Acceptable values for PowerState are **On**, **Off**, **PowerCycle**, and **Cancel**. Delay requires an integer in the range 0 to 2^{31}-1 (inclusive). The Delay parameter does not determine how long to delay before the action is applied. It instead determines whether the existing delay settings will be followed. If the Delay parameter is equal to zero, then the action will generally be performed immediately. If the Delay parameter is positive and non-zero, then the action will be delayed according to the existing delay settings. The table below lists the meaning of each accepted PowerState value and which delay value it follows.

Value	Description	Behavior with Non-Zero Delay Value	Behavior with Zero Delay Value
On	Turns the outlet on.	Turn on after PowerOnDelaySeconds seconds.	Turn on **immediately**.
Off	Turns the outlet off.	Turn off after PowerOffDelaySeconds seconds.	Turn off **immediately**.
PowerCycle	Turns the outlet off and back on.	Turn off after the reboot delay, then turn on after the reboot hold delay.	Turn off **immediately**, then turn on after the reboot hold delay.
Cancel	Cancel the pending outlet action.	Cancel the pending outlet action.	Cancel the pending outlet action.

2.6.10 Branch Collection: /redfish/v1/PowerEquipment/.../Branches

Identified branches will be listed here. A **branch** is a circuit connected to a circuit breaker. Supported URIs are of the form /redfish/v1/PowerEquipment/TransferSwitches/(ID)/Branches and /redfish/v1/PowerEquipment/RackPDUs/(ID)/Branches. The following properties are provided in the response to such a URI:

Property	Value	Description
@odata.id	/redfish/v1/PowerEquipment/<TransferSwitches/RackPDUs>/(ID)/Branches	
@odata.type	#CircuitCollection.CircuitCollection	
Members		Array
@odata.id	/redfish/v1/PowerEquipment/<TransferSwitches/RackPDUs>/(ID)/Branches/(BranchID)	
Members@odata.count	(Number of entries in Members array)	
Name	Branch Circuit Collection	

Branch

A **branch** is a circuit connected to a circuit breaker. For members of the **Members** array found at /redfish/v1/PowerEquipment/RackPDUs/(ID)/Branches or /redfish/v1/PowerEquipment/TransferSwitches/(ID)/Branches, the following properties within the Circuit schema are supported:

Property	Value	Description
@odata.id	/redfish/v1/PowerEquipment/<TransferSwit ches/RackPDUs>/(ID)/Branches/(BranchID)	
@odata.type	#Circuit.v1_5_0.Circuit	
Actions		Object
#Circuit.ResetMetrics		Object. Non-standard use. See Branch ResetMetrics on the next page.
target	/redfish/v1/PowerEquipment/<TransferSwit ches/RackPDUs>/(ID)/Branches/(BranchID)/ Actions/Circuit.ResetMetrics	
BreakerState	(BreakerState value)	**Off** if loss of load is detected. **Normal** otherwise.
CurrentAmps		Object
Reading	(Decimal value)	
Id	(BranchID)	
Links		Object
Outlets, see Outlet Collection: /redfish/v1/PowerEquipment/.../Outlets on page 27.		Array. Contains links to all outlets identified as being connected to this branch.
@odata.id	/redfish/v1/PowerEquipment/<TransferSwit ches/RackPDUs>/(ID)/Outlets/(OutletID)	
Name	Branch (Modified BranchID)	

Property	Value	Description
Status		Object
Health	OK	
State	Enabled	

Branch ResetMetrics

The Circuit.ResetMetrics action for a branch will not reset the metrics of said branch when used. Instead, triggering this action will attempt to reset the loss of load condition of the associated circuit breaker. Not all hardware is able to reset the loss of load condition remotely. Certain models will require physical intervention. Please verify that the hardware is capable of responding to such a reset attempt before relying on this functionality.

2.6.11 Metrics: /redfish/v1/PowerEquipment/.../Metrics

The PowerDistributionMetrics schema defines the properties available at URIs in the form of /redfish/v1/PowerEquipment/RackPDUs/(ID)/Metrics and /redfish/v1/PowerEquipment/TransferSwitches/(ID)/Metrics. The following properties within the PowerDistributionMetrics schema are supported:

Property	Value	Description
@odata.id	/redfish/v1/PowerEquipment/<TransferSwitches/ RackPDUs>/(ID)/Metrics	
@odata.type	#PowerDistributionMetrics.v1_3_0.PowerDistrib utionMetrics	
Actions		Object
#PowerDistributionMetrics.ResetMetrics		Object. Triggering this action resets the Metrics EnergykWh reading and the EnergykWh reading of all phases.
target	/redfish/v1/PowerEquipment/<TransferSwitches/ RackPDUs>/(ID)/Metrics/Actions/PowerDistributi onMetrics.ResetMetrics	
EnergykWh		Object
Reading	(Decimal value)	
Id	Metrics	
Name	Summary Metrics	
PowerWatts		Object
ApparentVA	(Decimal value)	
PowerFactor	(Decimal value)	
Reading	(Decimal value)	

2.7 EventService: /redfish/v1/EventService

Redfish EventService has partial support. Specifically, **push-style eventing** is supported while **server-sent events (SSE)-style eventing** is not supported. The only event format (EventFormatType) supported is **event** message objects (Event). **Metric report** message objects (MetricReport) can be specified in an EventDestination, but are not supported as TelemetryService is not supported. Events are sent via HTTP only and not HTTPS. Not all possible operations send an event. See Supported MessageIds on page 33 for a list of events that have some level of support.

Known issues: The delivery retry functionality may not work as expected at all times. Furthermore, in the event that a subscription should be deleted due to repeated delivery failures, that subscription may become inactive and yet not deleted.

The following properties of the EventService schema are supported:

Property	Value	Description
@odata.id	/redfish/v1/EventService	
@odata.type	#EventService.v1_5_0.EventService	
Actions		Object
#EventService.SubmitTestEvent		Object
target	/redfish/v1/EventService/Actions/Event Service.SubmitTestEvent	
DeliveryRetryAttempts	{Integer in range [1-3]}	**Default:** 3. Writable via PATCH.
DeliveryRetryIntervalSeconds	{Integer in range [30-180]}	**Default:** 30. Writable via PATCH.
EventFormatTypes	Event, MetricReport	Array
Id	EventService	
Name	Event Service	
RegistryPrefixes	AccountSecurity, Base, OpenBMC, ResourceEvent, StorageDevice, TaskEvent, Update	Array
ResourceTypes	Task	Not used as OriginOfCondition is not used.
SSEFilterPropertiesSupported		Object. SSE is not supported.
EventFormatType	true	
MessageId	true	
MetricReportDefinition	true	
OriginResource	false	
RegistryPrefix	true	
ResourceType	false	
ServiceEnabled	[true/false]	Writable via PATCH.
Status		Object
State	Enabled	
Subscriptions		Object
@odata.id	/redfish/v1/EventService/Subscriptions	

2.7.1 EventDestination Collection

Each EventService subscription is represented by an EventDestination. A list of all EventDestinations may be retreived via a GET request to the URI /redfish/v1/EventService/Subscriptions. New EventDestinations (and thus new subscriptions) can be created via a POST request to this URI. The following properties are supported in the response to a GET request to this URI:

Property	Value	Description
@odata.id	/redfish/v1/EventService/Subscriptions	
@odata.type	#EventDestinationCollection.EventDestinationCollection	
Members		Array
@odata.id	/redfish/v1/EventService/Subscriptions/(EventDestinationI D)	
Members@odata.count	(Number of entries in Members array)	
Name	Event Destination Collections	

The response to a POST request to this URI is normally a Message in response to the creation attempt.

2.7.2 EventDestination Creation

An EventDestination can be created by sending a POST request to the URI /redfish/v1/EventService/Subscriptions. The body of this POST request has a number of required and optional properties. See the table below for more information, where all properties are optional unless otherwise noted.

Property	Allowed Value (s)	Notes
Destination	{URI}	Required.
Protocol	Redfish	Required.
Context	{String}	Can modify via PATCH on existing EventDestination.
SubscriptionType	RedfishEvent	
EventFormatType	{Enum String}	**Value: Event** or **MetricReport**. TelemetryService, the generator of MetricReport events, is unsupported.
HttpHeaders	{Array of Objects}	Can modify via PATCH on existing EventDestination.
RegistryPrefixes	{Array of Enum Strings}	**Values: AccountSecurity**, **Base**, **OpenBMC**, **ResourceEvent**, **StrorageDevice**, **TaskEvent**, and/or **Update**. If provided, MessageIds must not be.
MessageIds	{Array of Strings}	Each String must be a message key only and may not include a registry prefix or any version information (e.g. **DriveInserted** instead of **StorageDevice.DriveInserted**). If provided, RegistryPrefixes must not be.
DeliveryRetryPolicy	{Enum String}	**Value: TerminateAfterRetries**, **SuspendRetries**, or **RetryForever**. Can modify via PATCH on existing EventDestination.
MetricReportDefinitions	{Array of Objects}	Each Object (key-value pair) must have the key **@odata.id**.
ResourceTypes	{Array of Enum Strings}	**Value:Task**.

See Supported MessageIds on the facing page for a list of supported MessageIds.

2.7.3 EventDestination Retrieval

Certain details of a specific EventDestination can be viewed at the URI /redfish/v1/EventService/Subscriptions/(EventDestinationID). The following properties from the EventDestination schema are supported:

Property	Value	Description
@odata.id	/redfish/v1/EventService/Subscrip tions/(EventDestinationID)	
@odata.type	#EventDestination.v1_7_0.EventDestination	
Context	{String}	Writable via PATCH.
DeliveryRetryPolicy	{Enum String}	Writable via PATCH. **Value: TerminateAfterRetries**, **SuspendRetries**, or **RetryForever**.
Destination	(String)	
EventFormatType	(Enum String)	**Value: Event** or **MetricReport**.
HttpHeaders	{Empty Array}	Writable via PATCH.
Id	(EventDestinationID)	
MessageIds		Array. Can be empty.
MetricReportDefinitions		Array. Can be empty.
@odata.id	(String)	
Name	Event Destination (EventDestinationID)	
Protocol	Redfish	
RegistryPrefixes		Array. Can be empty. **Possible array values: AccountSecurity**, **Base**, **OpenBMC**, **ResourceEvent**, **StorageDevice**, **TaskEvent**, and **Update**.
ResourceTypes		Array. Can be empty. **Possible array value: Task**.
SubscriptionType	RedfishEvent	

2.7.4 Supported MessageIds

The following MessageIds are supported and have some level of implementation.

Supported MessageId	Reporting Scenarios
AccountSecurity.InsufficientPrivilege	Attempting to transfer files to the card via SCP or access certain API URIs without admin privileges.
AccountSecurity.InvalidCredentials	Accessing certain API URIs without valid user credentials.
AccountSecurity.SuccessfulLogin	Successfully logging in via the CLI (serial, SSH, and SCP).
Base.CreateLimitReachedForResource	Provisioner firmware or config file uploaded through SCP when the maximum number of files have already been uploaded.
Base.UndeterminedFault	Various errors. Primarily with regard to transferring files to and from the IMD via SCP and the actions taken afterwards.
StorageDevice.DriveInserted	USB drive inserted.

Supported MessageId	Reporting Scenarios
StorageDevice.DriveRemoved	USB drive removed.
Update.UpdateInProgress	A firmware update is in progress.
Update.UpdateSuccessful	A firmware update has succeeded.

2.8 Chassis Collection: /redfish/v1/Chassis

The Chassis Collection URI will list a collection of IMD5s running firmware 7.0 or higher. Each chassis will have its own ID constructed from its MAC (Media Access Control) address after removing all **:** characters.

The following properties within the Chassis schema are supported:

Property	Value	Description
@odata.id	/redfish/v1/Chassis	
@odata.type	#ChassisCollection.ChassisCollection	
Members		Array
@odata.id	/redfish/v1/Chassis/(ChassisID)	ChassisID is the device's modified MAC address.
Members@odata.count	(Number of entries in Members array)	
Name	Chassis Collection	

2.8.1 Chassis

For elements of the /redfish/v1/Chassis/(ChassisID) Members array, the following properties within the Chassis schema are supported:

Property	Value	Description
@odata.id	/redfish/v1/Chassis/(ChassisID)	
@odata.type	#Chassis.v1_19_0.Chassis	
ChassisType	RackMount	
Id	(ChassisID)	Modified MAC address of device.
Name	Chassis (ChassisID)	
Oem		Object
Vertiv_Monitoring_VIDSecondaryDevs, see VID Secondary Devices on page 42.		List of VID secondary devices.
@odata.id	/redfish/v1/Chassis/(ChassisID)#/Oem/Vertiv_Monitoring_VIDSecondaryDevs	
Sensors, see Sensor Collection:/redfish/v1/Chassis/(ChassisID)/Sensors on the facing page.		List of 1-wire sensors.
@odata.id	/redfish/v1/Chassis/(ChassisID)/Sensors	

2.9 Sensor Collection: /redfish/v1/Chassis/(ChassisID)/Sensors

The Sensor Collection contains a collection of detected 1-wire sensor measurements. Multiple sensor measurements may be provided by a single physical sensor.

The following properties are supported:

Property	Value	Description
@odata.id	/redfish/v1/Chassis/(ChassisID)/Sensors	
@odata.type	#SensorCollection.SensorCollection	
Members		Array
@odata.id	/redfish/v1/Chassis/(ChassisID)/Sensors/(SensorID)	
Members@odata.count	(Number of entries in Members array)	
Name	Sensor Collection	

Below is an example response.


```
{
  "@odata.id": "/redfish/v1/Chassis/0002990b062a/Sensors",
  "@odata.type": "#SensorCollection.SensorCollection",
  "Members": [
      {
           "@odata.id":
 "/redfish/v1/Chassis/0002990b062a/Sensors/05000000ECD02112_Temperature"
      },
      {
           "@odata.id":
 "/redfish/v1/Chassis/0002990b062a/Sensors/05000000ECD02112_Humidity"
      },
      {
           "@odata.id":
 "/redfish/v1/Chassis/0002990b062a/Sensors/05000000ECD02112_Dewpoint"
      },
      {
           "@odata.id":
  "/redfish/v1/Chassis/0002990b062a/Sensors/05000000ECD02112_Airflow"
      }
   ],
    "Members@odata.count": 4,
    "Name": "Sensor Collection"
}
```


2.9.1 Sensors

The 1-wire sensors supported through Redfish protocol include the following:

- SN Temperature
- SN Door
- SN Leak Detector

- SN Relative Humidity
- A2D (Analog to Digital)
- Airflow (RTAFHD3)
- Temperature Sensor (remotetemp)
- Vertiv (Geist) Environmental Sensor 3 (GT3HD)
- Vertiv (Geist) Environmental Sensor (GTHD)

Some supported sensor devices will be displayed in Redfish as multiple sensors. This is because one physical sensor can provide multiple measurements, and each measurement is displayed as its own sensor. A physical sensor's serial number is included in the ID of all of its measurements. The general formula for determining the ID of a particular measurement provided by a particular sensor is (SensorSerialNumber)_(MeasurementType).

2.9.2 Temperature

URI: /redfish/v1/Chassis/(ChassisID)/Sensors/(SensorSerialNumber)_Temperature

Property	Value	Description
@odata.id	/redfish/v1/Chassis/(ChassisID)/Sensors/(SensorSerialNumber)_Temperature	
@odata.type	#Sensor.v1_5_0.Sensor	
Id	(SensorSerialNumber)_Temperature	
Name	(SensorName)	
Oem		Object
Vertiv_Monitoring_SensorMeasurement		Object
Thresholds		Array, see Sensor Thresholds on page 41 for more information.
Reading	(Decimal value)	Two digits after the decimal are provided.
ReadingType	Temperature	
ReadingUnits	<F/C>	
Status		Object
Health	OK	
State	Enabled	

2.9.3 Humidity

URI: /redfish/v1/Chassis/(ChassisID)/Sensors/(SensorSerialNumber)_Humidity

Property	Value	Description
@odata.id	/redfish/v1/Chassis/(ChassisID)/Sensors/(SensorSerialNumb er)_Humidity	
@odata.type	#Sensor.v1_5_0.Sensor	
Id	(SensorSerialNumber)_Humidity	

Property	Value	Description
Name	(SensorName)	
Oem		Object
Vertiv_Monitoring_SensorMeasurement		Object
Thresholds		Array, see Sensor Thresholds on page 41s for more information.
Reading	(Decimal value)	Two digits after the decimal are provided.
ReadingType	Humidity	
ReadingUnits	%	
Status		Object
Health	OK	
State	Enabled	

2.9.4 Dewpoint

URI: /redfish/v1/Chassis/(ChassisID)/Sensors/(SensorSerialNumber)_Dewpoint

Property	Value	Description
@odata.id	/redfish/v1/Chassis/(ChassisID)/Sensors/(SensorSerialNumb er)_ Dewpoint	
@odata.type	#Sensor.v1_5_0.Sensor	
Id	(SensorSerialNumber)_Dewpoint	
Name	(SensorName)	
Oem		Object
Vertiv_Monitoring_SensorMeasurement		Object
Thresholds		Array, see Sensor Thresholds on page 41 for more information.
Reading	(Decimal value)	Two digits after the decimal are provided.
ReadingType	Dewpoint	
ReadingUnits	<F/C>	
Status		Object
Health	OK	
State	Enabled	

2.9.5 Airflow

URI: /redfish/v1/Chassis/(ChassisID)/Sensors/(SensorSerialNumber)_Airflow

Property	Value	Description
@odata.id	/redfish/v1/Chassis/(ChassisID)/Sensors/(SensorSerialNumber)_Airflow	
@odata.type	#Sensor.v1_5_0.Sensor	
Id	(SensorSerialNumber)_Airflow	
Name	(SensorName)	
Oem		Object
Vertiv_Monitoring_SensorMeasurement		Object
Thresholds		Array, see Sensor Thresholds on page 41 for more information.
Reading	(Decimal value)	Two digits after the decimal are provided.
ReadingType	AirFlow	
ReadingUnits		Not applicable
Status		Object
Health	OK	
State	Enabled	

2.9.6 Leak Detection

URI: /redfish/v1/Chassis/(ChassisID)/Sensors/(SensorSerialNumber)_Leak_Detection

Property	Value	Description
@odata.id	/redfish/v1/Chassis/(ChassisID)/Sensors/(SensorSerialNumber)_Leak_Detection	
@odata.type	#Sensor.v1_5_0.Sensor	
Id	(SensorSerialNumber)_Leak_Detection	
Name	(SensorName)	
Oem		Object
Vertiv_Monitoring_SensorMeasurement		Object
Thresholds		Array, see Sensor Thresholds on page 41 for more information.
Reading	<Wet/Ok>	
ReadingType	Leak Detection	
ReadingUnits		Not applicable

Property	Value	Description
Status		Object
Health	OK	
State	Enabled	

2.9.7 Fault Detection

URI: /redfish/v1/Chassis/(ChassisID)/Sensors/(SensorSerialNumber)_Fault_Detection

Property	Value	Description
@odata.id	/redfish/v1/Chassis/(ChassisID)/Sensors/(SensorSerialNumber)_Fault_Detection	
@odata.type	#Sensor.v1_5_0.Sensor	
Id	(SensorSerialNumber)_Fault_Detection	
Name	(SensorName)	
Oem		Object
Vertiv_Monitoring_SensorMeasurement		Object
Thresholds		Array, see Sensor Thresholds on page 41 for more information.
Reading	<Fault/Ok>	
ReadingType	Fault Detection	
ReadingUnits		Not applicable
Status		Object
Health	OK	
State	Enabled	

2.9.8 Door

SN Door sensor can monitor up to two doors. The sensors will be displayed as Door_1 and/or Door_2.

URI: /redfish/v1/Chassis/(ChassisID)/Sensors/(SensorSerialNumber)_Door_<1/2>

Property	Value	Description
@odata.id	/redfish/v1/Chassis/(ChassisID)/Sensors/(SensorSerialNumb er)_Door_<1/2>	
@odata.type	#Sensor.v1_5_0.Sensor	
Id	(SensorSerialNumber)_Door_<1/2>	
Name	SN Door	
Oem		Object
Vertiv_Monitoring_SensorMeasurement		Object

Property	Value	Description
Thresholds		Array, see Sensor Thresholds on the facing page for more information.
Reading	<Open/Closed>	
ReadingType	Door	
ReadingUnits		Not applicable
Status		Object
Health	OK	
State	Enabled	

2.9.9 A2D Reading Types (Modes)

A2D sensors may be configured to operate in different modes. Configuration of A2D sensors is not available through the Redfish API. An example of the supported properties for an A2D sensor configured in the Custom (Voltage mode) is provided below:

Custom (Voltage Mode)

URI: /redfish/v1/Chassis/(ChassisID)/Sensors/(SensorSerialNumber)_customVoltage

Property	Value	Description
@odata.id	/redfish/v1/Chassis/(ChassisID)/Sensors/(SensorSerialNumber)_customVoltage	
@odata.type	#Sensor.v1_5_0.Sensor	
Id	(SensorSerialNumber)_customVoltage	
Name	A2D	
Oem		Object
Vertiv_Monitoring_SensorMeasurement		Object
Thresholds		Array, see Sensor Thresholds on the facing page for more information.
Reading	(Decimal value)	Value based on A2D configuration.
ReadingType	Custom (Voltage Mode)	**NOTE: Voltage mode is a literal string.**
ReadingUnits	V	Units based on A2D configuration.
Status		Object
Health	OK	
State	Enabled	

2.9.10 Sensor Thresholds

If an alarm or warning has been defined for a particular measurement of a sensor, then certain additional information will be populated within the Thresholds Object returned for that sensor. These alarms or warning may not be created, deleted, or modified using the Redfish API. See below for an example response which includes sensor threshold data.


```
{
  "@odata.id": "/redfish/v1/Chassis/0002990b062a/Sensors/05000000ECD02112_Airflow",
"@odata.type": "#Sensor.v1_5_0.Sensor",
  "Id": "05000000ECD02112_Airflow",
  "Name": "RTAFHD3",
  "Oem": {
    "   Vertiv_Monitoring_SensorMeasurement": {
    "Thresholds": [
         {
           "Activation": "Increasing",
           "ClearingDelay": 1,
           "DwellTime": "PT1S",
           "InvertionTime": false,
           "LatchingMode": true,
           "Reading": 34.0,
           "Severity": "UpperCritical",
           "Targets": [
               {
                  "Delay": 0,
                  "Enabled": false,
                  "Name": "outletD02A060B990200C3-0",
                  "Repeat": 1,
                  "Target": "Outlet 1"
               },
               {
                  "Delay": 0,
                  "Enabled": true,
                  "Name": "trap0",
                  "Repeat": 2,
                  "Target": "10.xx.xx.xx"
                }
              ],
              "TripDelay": 1,
              "ValidTime": "-TWTF--00:00-23:45"
            }
         ]
      }
   },
   "Reading": 36.0,
   "ReadingType": "AirFlow",
   "ReadingUnits": "",
   "Status": { "Health": "OK", "State": "Enabled" }
}
```

2.10 VID Secondary Devices

When other devices running firmware version 7.0 or higher are connected together to form a VID (Vertiv Intelligence Director) array, those devices will be displayed in an OEM Object under the **/redfish/v1/Chassis/(ChassisID)** endpoint, where the **ChassisID** is the MAC address of the VID director (the **primary** unit). Multiple types of devices may be displayed here. These device types are RackPDUs, TransferSwitches, and Vertiv™ RDU202s (Watchdog).

IMPORTANT! Only secondary devices running 7.0 firmware or higher will be listed. Likewise, only external sensors connected to devices running 7.0 firmware or higher will be listed.

Property	Value	Description
@odata.id	/redfish/v1/Chassis/(ChassisID)#/Oem/Vertiv_Monitoring_V IDSecondaryDevs	
@odata.type	#OemChassis.v1_0_0.VIDSecondaryDevs	
ExternalSensors		Array. Can be empty.
@odata.id	(Sensors URI of secondary device if not a Liebert® RDU202 unit)	
RDU202		Array. Can be empty.
@odata.id	(Sensors URI of secondary device if a Liebert® RDU202 unit)	
RackPDUs		Array. Can be empty.
@odata.id	(RackPDU URI of secondary device if a RackPDU unit)	
TransferSwitches		Array. Can be empty.
@odata.id	(TransferSwitch URI of secondary device if a TransferSwitch unit)	

Below is an example of a response to the **/redfish/v1/Chassis/000068101296** URI showing the response when an external sensor is connected to a device in the VID array.


```
{
  "@odata.id": "/redfish/v1/Chassis/000068101296", "@odata.type": "#Chassis.v1_19_
0.Chassis", "ChassisType": "RackMount",
  "Id": "000068101296",
  "Name": "Chassis 000068101296",
  "Oem": {
        "Vertiv_Monitoring_VIDSecondaryDevs": {
        "@odata.id":
"/redfish/v1/Chassis/000068101296#/Oem/Vertiv_Monitoring_VIDSecondaryDevs",
        "@odata.type": "#OemChassis.v1_0_0.VIDSecondaryDevs",
        "ExternalSensors": [
         {
             "@odata.id":
"https://10.20.35.158:17078/redfish/v1/Chassis/0000681013ee/Sensors"
         }
      ],
      "RDU202": [],
      "RackPDUs": [
         {
            "@odata.id":
"https://10.20.35.158:17078/redfish/v1/PowerEquipment/RackPDUs/9EEE1310680000C3"
         }
       ],
```

```
        "TransferSwitches": []
    }
},
"Sensors": {
     "@odata.id": "/redfish/v1/Chassis/000068101296/Sensors"
    }
}
```


Notice from the example above that port-forwarding is used to permit access to the Redfish™ implementation on devices within the VID array. This access should be used carefully to prevent configuring a device in such a way that the VID functionality is impeded. See Vertiv Intelligence Director documentation for more information on configuration and accessing secondary units.

2.10.1 Sensors Connected to Secondary Devices

Any 1-wire sensors connected to a secondary device of type IMD5 running 7.0 firmware or higher may be accessed via the Sensors URI of the secondary device. For example, given the scenario where the VID director unit has a ChassisID of 000068101296 (MAC address of 00:00:68:10:12:96), the measurements of sensors connected to a secondary unit with a ChassisID of 0000681013ee may be accessed using the IP address of the director unit and the port assigned to that secondary unit at the URI /redfish/v1/Chassis/0000681013ee/Sensors.

Please see below for an example response for the Sensors URI of a secondary device with ChassisID 0002990b062a:


```
{
   "@odata.id": "/redfish/v1/Chassis/0002990b062a/Sensors",
   "@odata.type": "#SensorCollection.SensorCollection",
   "Members": [
       {
         "@odata.id":
"/redfish/v1/Chassis/0002990b062a/Sensors/05000000ECD02112_Temperature"
       },
       {
         "@odata.id":
"/redfish/v1/Chassis/0002990b062a/Sensors/05000000ECD02112_Humidity"
       },
       {
         "@odata.id":
"/redfish/v1/Chassis/0002990b062a/Sensors/05000000ECD02112_Dewpoint"
        },
        {
         "@odata.id":
"/redfish/v1/Chassis/0002990b062a/Sensors/05000000ECD02112_Airflow"
        }
     ],
        "Members@odata.count": 4,
        "Name": "Sensor Collection"
}
```


Note that the sensor measurements are displayed as described in Sensor Collection:/redfish/v1/Chassis/(ChassisID)/Sensors on page 35. Requesting the URI for the Airflow measurement would provide a response like the following:

```
{
   "@odata.id": "/redfish/v1/Chassis/0002990b062a/Sensors/05000000ECD02112_Airflow",
   "@odata.type": "#Sensor.v1_5_0.Sensor",
   "Id": "05000000ECD02112_Airflow",
   "Name": "RTAFHD3",
   "Oem": {
   "Vertiv_Monitoring_SensorMeasurement": {
       "Thresholds": []
     }
   },
   "Reading": 42.0,
   "ReadingType": "AirFlow",
   "ReadingUnits": "",
   "Status": {
      "Health": "OK",
      "State": "Enabled"
    }
}
```

This page intentionally left blank

Appendices

Appendix A: Technical Support and Contacts

A.1 Technical Support/Service in the United States

Vertiv Group Corporation

24x7 dispatch of technicians for all products.

1-800-543-2378

Liebert® Thermal Management Products

1-800-543-2378

Liebert® Channel Products

1-800-222-5877

Liebert® AC and DC Power Products

1-800-543-2378

A.2 Locations

United States

Vertiv Headquarters

505 N Cleveland Ave

Westerville, OH 43082

Europe

Via Leonardo Da Vinci 8 Zona Industriale Tognana

35028 Piove Di Sacco (PD) Italy

Asia

7/F, Dah Sing Financial Centre

3108 Gloucester Road, Wanchai

Hong Kong

This page intentionally left blank

Connect with Vertiv on Social Media

 https://www.facebook.com/vertiv/

 https://www.instagram.com/vertiv/

 https://www.linkedin.com/company/vertiv/

 https://www.x.com/Vertiv/

Vertiv.com | Vertiv Headquarters, 505 N Cleveland Ave, Westerville, OH 43082 USA

SL-80313_REVA_10-25