DELL Technologies

White Paper

Telemetry Metric Report Definition Explained

This Whitepaper explains different parameters of Metric Report Definitions that are configurable using Redfish, Server Configuration Profile (SCP), and iDRAC Graphical User Interface for the desired telemetry report data and the streaming behavior.

August 2021

Revisions

Date	Description
August 2021	Initial release

Acknowledgments

Authors: Ankita Goyal, Cyril Jose, Sankara Gara, Michael E Brown

Table of contents

Executive summary

With iDRAC9 v4.00.00.00, Dell EMC introduced the Telemetry Streaming solution for PowerEdge servers. The feature enables IT managers to integrate advanced server hardware operations telemetry into their existing analytics solutions. iDRAC9 v4.40 introduces custom configuration of metric report definition through Redfish and Server Configuration Profile methods.

Custom telemetry configuration is DMTF Redfish compliant and allows consumers to pick and choose the telemetry report configuration settings and the metrics included in the report in much more granular detail. This paper explores and details the configurable properties of metric report definition.

1. Introduction

Metric Report Definitions defines the backbone of telemetry metric reports. The paper describes definitions of the properties in Metric Report Definitions. Custom telemetry configuration is DMTF Redfish compliant, and this paper discusses the iDRAC behavior for each configuration options.

1.1 Terms and Definition

To this technical white paper, following terms and definitions apply:

MRD – Metric Report Definitions

MR – Metric Report

MD – Metric Definitions

1.2 Prerequisites

iDRAC Datacenter license is required to collect telemetry metric reports.

2. Metric Report Definition Properties

2.1 Configurable Properties

MetricReportDefinitionEnabled – This property is used to enable or disable the report. After uploading the Report Definition, perform PATCH method on this property to dynamically enable or disable the report. *Default is "false"*.

MetricReportDefinitionType – Defines the type of the metric report that is generated from the metric report definition.

Periodic	This is the default for any new reports that are uploaded with unspecified type. This type of report must have a "Schedule" object with a "RecurrenceInterval" property. The report is repeatedly generated on a periodic basis as requested in the "Schedule/RecurrenceInterval" property, with the report being generated at the expiry of the time period specified by "Schedule/RecurrenceInterval" property. When metric reports are generated, the report includes metrics greater than the previous report timestamp up to and including the new report timestamp. If a "ReportTimespan" is also present, the resulting metric value list includes the set-wise union of values that are covered by ReportTimespan and metric values with timestamps in the interval of previous ReportTimestamp to current ReportTimestamp.
OnChange	This report is generated anytime a referenced metric change. To prevent overloading several subsystems with reports, the minimum time between reports is 10 seconds. If multiple metrics are changed during the 10 seconds "cooling off" period, all metrics that have changed up until the report generation time are included in the report. For "OnChange" reports, a non-null "TimeSpan" property is mandatory, the "SuppressRepeatedMetricValue" property is required to be set to true. The report shall contain all metrics values newer than "Report Timestamp – TimeSpan" up until the Report Timestamp. Additionally, "OnChange" requires HeartBeatInterval and schedule to be cleared.
OnRequest	This report is generated when there is a HTTP GET in the report. The report contains all metrics values newer than Now() - ReportTimeSpan. Other Requirements: • ReportUpdates field is IGNORED in input. Output will be fixed to AppendWrapsWhenFull • ReportActions field is IGNORED in input. Output will be fixed to LogToMetricReportsCollection • ReportTimeSpan must have a value greater than 0 • Schedule property must be either not present in input, or the RecurrenceInterval must be 'null' or equivalent to any variant of 'PT0S'.

ReportActions – This property specifies what should happen when a report is generated per the MetricReportDefinitionType, above. The property is an ARRAY, containing zero or more of the following settings. Note that if no settings are specified, it disables the report. Default is ["LogToMetricReportsCollection", "RedfishEvent"]

LogToMetricReportsCollection	The generated metric report is added to the Metric Reports collection with an “Id” matching the report definition.
RedfishEvent	The generated metric report is sent as a “Metric Report” event type to all subscribers: SSE Subscribers, POST Subscribers, and RSYSLOG subscribers. The report is not saved to the metric reports collection unless “LogToMetricReportsCollection” is also specified. Note that the individual subscription types may further have facilities to filter which reports are sent. For example, SSE can query on only specific reports. This is property provides base enablement but does not force this report into streams that have filters.

ReportUpdates – This property controls how subsequent reports are handled after the first report is generated.

AppendStopsWhenFull	Each time the report is generated, the new metric values are added to the end of the MetricValues array. The report can contain up to an “AppendLimit” (not user configurable, described below) number of metric values. Once the maximum number of metric values have been added, no more will be added to the report and the MetricReportDefinition is automatically disabled. The user must PATCH the MetricReportDefinition back to enable the report to restart the report generation. Reports will not stream after they have stopped. The detection of hitting append limit is ‘asynchronous’, done periodically as a background task, so streaming may not immediately stop, but should stop within 15 minutes of hitting the limit.
AppendWrapsWhenFull	When the report is created, new Metrics are added to the MetricValues array, keeping existing entries. Once there are more than “AppendLimit” number of metric values in the MetricValues array, older entries above the AppendLimit number are dropped from the report.
NewReport	A new report is added, with a name generated taking the report definition name as a base and adding a dash, then the current date/time. Since this implies that many older reports are available under the older names, this can present a challenge to iDRAC, so defining this specific behavior for allows: • Deleting any report that is already processed and is not needed any more. • Automatic deletion of reports that are older than the last three, as iDRAC will keep at most three completed metric reports per report definition.
Overwrite	Overwrites the older report with a report of the same name containing newer data. This is the default.

SuppressRepeatedMetricValue – If Enabled, new metrics are added to the MetricValue array if they are different from the last reported metric. For a single report, if a metric value does not change, it is not added again.

Depending on the specific report configuration (“ReportUpdates”, and “MetricReportDefinitionType”), the report level behavior differs. SuppressRepeatedMetricValue prevents addition of consecutive duplicate metric values to a specific metric value stream. Default is “Disabled” for new custom reports.

MetricReportHeartbeatInterval – For metric reports that have SuppressRepeatedMetricValues (“suppress”) set to true, some metrics may not change for a very long time and fail to show up in many reports. This is expected, but sometimes end users may wish to have periodic ‘reminders’ (conceptually) of which metrics are valid for this report. The heartbeat setting will periodically force a report that contains at least one value for every currently valid metric. This property is a Redfish Duration and should always be greater than or equal to the Recurrence Interval.

Note that this will only unsuppress nonaggregated metrics. Any metrics with a CollectionFunction are not affected by this setting. The way this duration works is that every time that is evenly divisible by the duration constitutes a conceptual trigger point where the ‘next’ report generated after that time will be a heartbeat report. Note that heartbeatinterval is only valid for “Periodic” type reports. Default is 0 (disabled). Setting to NULL or 0 will disable this feature.

Schedule – This property is the DMTF standard Schedule property and specifies the recurrence for the report. The “RecurrenceInterval” property of the schedule object specifies a Redfish Duration string. This property is valid for “Periodic” reports only. When set, reports are generated at the specified interval**.**

RecurrenceInterval value of 0 results in a nonrepeating, single report that may or may not have data depending on availability of data in the database.

ReportTimespan – This property specifies the duration of the report. This attribute must be set if the report is only required for a specific period or time span. This property setting is not applicable to “SerialLog” report where the report (log) continuity is expected.

Metrics – An array property that lists the metric IDs to be included in the metric report which the metric report definition defines. Each metric ID can have following configurable attributes:

CollectionFunction	The Redfish standard defines a way to reduce the raw number of data points sent by aggregating metrics using predefined metric functions that are applied during collection. The possible values are “Average”, “Maximum”, “Minimum”, and “Summation”. Default is ‘null’. When applying a collection function, the function is applied across a time interval and one single value is added to the report. If specified, CollectionDuration must also be present.
CollectionDuration	Specifies the duration over which the function is computed. Specified according to the Duration format in the Redfish schema supplement. Default is ‘null’. If specified, CollectionFunction must also be specified.
CollectionTimeScope	Interval – The timestamp of the resulting MetricValue is the end of the time interval. This is used and only valid when CollectionFunction and CollectionDuration are specified. Point – Metric values are point in time values (implies no collection function applied). This is the default.

MetricId	All the user to select all metrics with the corresponding matched ID to add to the report. All selection criteria here and below are considered (“AND”-ed) together to select metrics.
MetricProperties	Array of URIs of properties included in this metric. This property is not implemented.
OEM/Dell/Source	Allows you to to specify the data source to include. This is logically “AND”-ed with the metric id and FQDD, if specified. It can be specified by itself to request all MetricIds from that source be included.
OEM/Dell/FQDD	Allows you to to specify an FQDD filter. This is logically “AND”-ed with all other selection criteria. It can be specified by itself to request all MetricIds that are reported for that FQDD.
OEM/Dell/CustomLabel	The value of the customLabel field is used as the label for that metric in the metric report and overrides the default label.

Read Only Properties

AppendLimit – This value is a read only value that is determined by the Telemetry Service, and will be added as output to any Metric Report Definition. This parameter represents the maximum number of metrics in a report. The default value is 2400.

Status – This read-only property reflects the current state of the report definition. Values are “Enabled” or “Disabled” and reflect the MetricReportDefinitionEnabled value. If the report is “AppendStopsWhenFull”, then this is updated to “Disabled” as well as the MetricReportDefinitionEnabled property.

Read only OEM attributes are listed below.

OEM/Dell/Digest – The attribute allows you to Identify out of band changes in custom MRD outside the influence of the component that created it. Digest represent a consistent hash of the editable fields in metric report definition and remain constant if the definition remains intact. Digest changes with updates in the report definition. Digest also change with firmware updates if the updated version includes changes to editable metric report definition fields.

OEM/Dell/iDRACFirmwareVersion – Represents the current installed version of the iDRAC firmware.

3. Sample Metric Report Definitions

3.1 *Report with all metrics relevant for CPU.Socket.% Device*


```
{
    "@odata.type": "#MetricReportDefinition.v1_3_3.MetricReportDefinition",
    "@odata.context": "/redfish/v1/$metadata#MetricReportDefinition.MetricReportDefinition",
    "@odata.id": "/redfish/v1/TelemetryService/MetricReportDefinitions/CPUUberReport",
    "Id": "CPUUberReport",
    "Name": "All CPU metrics",
    "Description": "A report with all datapoints relevant to a specific device. (All CPU Socke
ts)",
    "AppendLimit": 2400,
    "MetricReportDefinitionEnabled": true,
    "MetricReportDefinitionType": "Periodic",
    "MetricReportHeartbeatInterval": "PT0H0M0S",
    "SuppressRepeatedMetricValue": false,
    "ReportTimespan": "PT0H1M0S",
    "ReportUpdates": "Overwrite",
    "MetricReportDefinitionType@Redfish.AllowableValues": [
        "Periodic",
        "OnChange",
        "OnRequest"
    ],
    "MetricReportDefinitionType@Redfish.AllowableValues@odata.count": 3,
    "ReportUpdates@Redfish.AllowableValues": [
        "AppendStopsWhenFull",
        "AppendWrapsWhenFull",
        "NewReport",
        "Overwrite"
    ],
    "ReportUpdates@Redfish.AllowableValues@odata.count": 4,
    "ReportActions": [
        "LogToMetricReportsCollection",
        "RedfishEvent"
    ],
    "ReportActions@odata.count": 2,
    "ReportActions@Redfish.AllowableValues": [
        "LogToMetricReportsCollection",
        "RedfishEvent"
    ],
    "ReportActions@Redfish.AllowableValues@odata.count": 2,
    "Status": {
        "State": "Enabled"
    },
    "Wildcards": [],
```

```
    "Wildcards@odata.count": 0,
    "Schedule": {
        "RecurrenceInterval": "PT0H1M0S"
    },
    "MetricReport": {
        "@odata.id": "/redfish/v1/TelemetryService/MetricReports/CPUUberReport"
    },
    "Metrics": [
        {
            "MetricId": null,
            "MetricProperties": [],
            "MetricProperties@odata.count": 0,
            "CollectionFunction": null,
            "CollectionDuration": null,
            "CollectionTimeScope": "Point",
            "Oem": {
                "Dell": {
                    "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinit
ion",
                    "CustomLabel": null,
                    "FQDD": "CPU.Socket.%",
                    "Source": null
                }
            }
        }
    ],
    "Metrics@odata.count": 1,
    "Links": {
        "Triggers": [],
        "Triggers@odata.count": 0
    },
    "Oem": {
        "Dell": {
            "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinition",
            "Digest": "a3ea958c8bfbc2015453caaf3d8422cb37b5f8ec1cae97c013cd0164caf0e997",
            "iDRACFirmwareVersion": "5.10.00.00"
        }
    }
}
```


3.2 *Report with Temperature and FAN RPM Readings Metrics*


```
{
    "@odata.type": "#MetricReportDefinition.v1_3_3.MetricReportDefinition",
    "@odata.context": "/redfish/v1/$metadata#MetricReportDefinition.MetricReportDefinition",
```

```
    "@odata.id": "/redfish/v1/TelemetryService/MetricReportDefinitions/TemperatureAndRPMReadin
g",
    "Id": "TemperatureAndRPMReading",
    "Name": "All Temeperature and RPM Readings",
    "Description": "A report with all relevant thermal metrics - Temperature reading and FAN R
PM readings",
    "AppendLimit": 2400,
    "MetricReportDefinitionEnabled": true,
    "MetricReportDefinitionType": "Periodic",
    "MetricReportHeartbeatInterval": "PT0H0M0S",
    "SuppressRepeatedMetricValue": false,
    "ReportTimespan": "PT0H1M0S",
    "ReportUpdates": "Overwrite",
    "MetricReportDefinitionType@Redfish.AllowableValues": [
        "Periodic",
        "OnChange",
        "OnRequest"
    ],
    "MetricReportDefinitionType@Redfish.AllowableValues@odata.count": 3,
    "ReportUpdates@Redfish.AllowableValues": [
        "AppendStopsWhenFull",
        "AppendWrapsWhenFull",
        "NewReport",
        "Overwrite"
    ],
    "ReportUpdates@Redfish.AllowableValues@odata.count": 4,
    "ReportActions": [
        "LogToMetricReportsCollection",
        "RedfishEvent"
    ],
    "ReportActions@odata.count": 2,
    "ReportActions@Redfish.AllowableValues": [
        "LogToMetricReportsCollection",
        "RedfishEvent"
    ],
    "ReportActions@Redfish.AllowableValues@odata.count": 2,
    "Status": {
        "State": "Enabled"
    },
    "Wildcards": [],
    "Wildcards@odata.count": 0,
    "Schedule": {
        "RecurrenceInterval": "PT0H1M0S"
    },
    "MetricReport": {
        "@odata.id": "/redfish/v1/TelemetryService/MetricReports/TemperatureAndRPMReading"
    },
    "Metrics": [
```

```
        {
            "MetricId": "TemperatureReading",
            "MetricProperties": [],
            "MetricProperties@odata.count": 0,
            "CollectionFunction": null,
            "CollectionDuration": null,
            "CollectionTimeScope": "Point",
            "Oem": {
                "Dell": {
                    "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinit
ion",
                    "CustomLabel": null,
                    "FQDD": null,
                    "Source": null
                }
            }
        },
        {
            "MetricId": "RPMReading",
            "MetricProperties": [],
            "MetricProperties@odata.count": 0,
            "CollectionFunction": null,
            "CollectionDuration": null,
            "CollectionTimeScope": "Point",
            "Oem": {
                "Dell": {
                    "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinit
ion",
                    "CustomLabel": null,
                    "FQDD": null,
                    "Source": null
                }
            }
        }
    ],
    "Metrics@odata.count": 2,
    "Links": {
        "Triggers": [],
        "Triggers@odata.count": 0
    },
    "Oem": {
        "Dell": {
            "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinition",
            "Digest": "82f47e45c966bdbdf33cd904756178f934fb86ec55d4dfcdae2c0b8e129be2d8",
            "iDRACFirmwareVersion": "5.10.00.00"
        }
    }
}
```

3.3 *Report with System Usage Aggregations*


```
{
    "@odata.type": "#MetricReportDefinition.v1_3_3.MetricReportDefinition",
    "@odata.context": "/redfish/v1/$metadata#MetricReportDefinition.MetricReportDefinition",
    "@odata.id": "/redfish/v1/TelemetryService/MetricReportDefinitions/SystemUsageAggregations
",
    "Id": "SystemUsageAggregations",
    "Name": "System usage metrics",
    "Description": "A report with average, maximum and minimum system usage metrics",
    "AppendLimit": 2400,
    "MetricReportDefinitionEnabled": true,
    "MetricReportDefinitionType": "Periodic",
    "MetricReportHeartbeatInterval": "PT0H0M0S",
    "SuppressRepeatedMetricValue": false,
    "ReportTimespan": "PT0H1M0S",
    "ReportUpdates": "Overwrite",
    "MetricReportDefinitionType@Redfish.AllowableValues": [
        "Periodic",
        "OnChange",
        "OnRequest"
    ],
    "MetricReportDefinitionType@Redfish.AllowableValues@odata.count": 3,
    "ReportUpdates@Redfish.AllowableValues": [
        "AppendStopsWhenFull",
        "AppendWrapsWhenFull",
        "NewReport",
        "Overwrite"
    ],
    "ReportUpdates@Redfish.AllowableValues@odata.count": 4,
    "ReportActions": [
        "LogToMetricReportsCollection",
        "RedfishEvent"
    ],
    "ReportActions@odata.count": 2,
    "ReportActions@Redfish.AllowableValues": [
        "LogToMetricReportsCollection",
        "RedfishEvent"
    ],
    "ReportActions@Redfish.AllowableValues@odata.count": 2,
    "Status": {
        "State": "Enabled"
    },
    "Wildcards": [],
    "Wildcards@odata.count": 0,
    "Schedule": {
        "RecurrenceInterval": "PT0H1M0S"
    },
```

```
    "MetricReport": {
        "@odata.id": "/redfish/v1/TelemetryService/MetricReports/SystemUsageAggregations"
    },
    "Metrics": [
        {
            "MetricId": "CPUUsage",
            "MetricProperties": [],
            "MetricProperties@odata.count": 0,
            "CollectionFunction": "Average",
            "CollectionDuration": "PT0H1M0S",
            "CollectionTimeScope": "Interval",
            "Oem": {
                "Dell": {
                    "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinit
ion",
                    "CustomLabel": null,
                    "FQDD": null,
                    "Source": null
                }
            }
        },
        {
            "MetricId": "CPUUsage",
            "MetricProperties": [],
            "MetricProperties@odata.count": 0,
            "CollectionFunction": "Maximum",
            "CollectionDuration": "PT0H1M0S",
            "CollectionTimeScope": "Interval",
            "Oem": {
                "Dell": {
                    "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinit
ion",
                    "CustomLabel": null,
                    "FQDD": null,
                    "Source": null
                }
            }
        },
        {
            "MetricId": "CPUUsage",
            "MetricProperties": [],
            "MetricProperties@odata.count": 0,
            "CollectionFunction": "Minimum",
            "CollectionDuration": "PT0H1M0S",
            "CollectionTimeScope": "Interval",
            "Oem": {
                "Dell": {
```

```
                    "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinit
ion",
                    "CustomLabel": null,
                    "FQDD": null,
                    "Source": null
                }
            }
        },
        {
            "MetricId": "IOUsage",
            "MetricProperties": [],
            "MetricProperties@odata.count": 0,
            "CollectionFunction": "Average",
            "CollectionDuration": "PT0H1M0S",
            "CollectionTimeScope": "Interval",
            "Oem": {
                "Dell": {
                    "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinit
ion",
                    "CustomLabel": null,
                    "FQDD": null,
                    "Source": null
                }
            }
        },
        {
            "MetricId": "IOUsage",
            "MetricProperties": [],
            "MetricProperties@odata.count": 0,
            "CollectionFunction": "Maximum",
            "CollectionDuration": "PT0H1M0S",
            "CollectionTimeScope": "Interval",
            "Oem": {
                "Dell": {
                    "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinit
ion",
                    "CustomLabel": null,
                    "FQDD": null,
                    "Source": null
                }
            }
        },
        {
            "MetricId": "IOUsage",
            "MetricProperties": [],
            "MetricProperties@odata.count": 0,
            "CollectionFunction": "Minimum",
            "CollectionDuration": "PT0H1M0S",
```

```
            "CollectionTimeScope": "Interval",
            "Oem": {
                "Dell": {
                    "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinit
ion",
                    "CustomLabel": null,
                    "FQDD": null,
                    "Source": null
                }
            }
        },
        {
            "MetricId": "MemoryUsage",
            "MetricProperties": [],
            "MetricProperties@odata.count": 0,
            "CollectionFunction": "Average",
            "CollectionDuration": "PT0H1M0S",
            "CollectionTimeScope": "Interval",
            "Oem": {
                "Dell": {
                    "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinit
ion",
                    "CustomLabel": null,
                    "FQDD": null,
                    "Source": null
                }
            }
        },
        {
            "MetricId": "MemoryUsage",
            "MetricProperties": [],
            "MetricProperties@odata.count": 0,
            "CollectionFunction": "Maximum",
            "CollectionDuration": "PT0H1M0S",
            "CollectionTimeScope": "Interval",
            "Oem": {
                "Dell": {
                    "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinit
ion",
                    "CustomLabel": null,
                    "FQDD": null,
                    "Source": null
                }
            }
        },
        {
            "MetricId": "MemoryUsage",
            "MetricProperties": [],
```

```
            "MetricProperties@odata.count": 0,
            "CollectionFunction": "Minimum",
            "CollectionDuration": "PT0H1M0S",
            "CollectionTimeScope": "Interval",
            "Oem": {
                "Dell": {
                    "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinit
ion",
                    "CustomLabel": null,
                    "FQDD": null,
                    "Source": null
                }
            }
        },
        {
            "MetricId": "AggregateUsage",
            "MetricProperties": [],
            "MetricProperties@odata.count": 0,
            "CollectionFunction": "Average",
            "CollectionDuration": "PT0H1M0S",
            "CollectionTimeScope": "Interval",
            "Oem": {
                "Dell": {
                    "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinit
ion",
                    "CustomLabel": null,
                    "FQDD": null,
                    "Source": null
                }
            }
        },
        {
            "MetricId": "AggregateUsage",
            "MetricProperties": [],
            "MetricProperties@odata.count": 0,
            "CollectionFunction": "Maximum",
            "CollectionDuration": "PT0H1M0S",
            "CollectionTimeScope": "Interval",
            "Oem": {
                "Dell": {
                    "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinit
ion",
                    "CustomLabel": null,
                    "FQDD": null,
                    "Source": null
                }
            }
        },
```

```
        {
            "MetricId": "AggregateUsage",
            "MetricProperties": [],
            "MetricProperties@odata.count": 0,
            "CollectionFunction": "Minimum",
            "CollectionDuration": "PT0H1M0S",
            "CollectionTimeScope": "Interval",
            "Oem": {
                "Dell": {
                    "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinit
ion",
                    "CustomLabel": null,
                    "FQDD": null,
                    "Source": null
                }
            }
        }
    ],
    "Metrics@odata.count": 12,
    "Links": {
        "Triggers": [],
        "Triggers@odata.count": 0
    },
    "Oem": {
        "Dell": {
            "@odata.type": "#DellMetricReportDefinition.v1_1_0.DellMetricReportDefinition",
            "Digest": "bffb0c579c53d24c7bf3fb8fdaf6c0ff4ccd487072bab02c16c7864827abe420",
            "iDRACFirmwareVersion": "5.10.00.00"
        }
    }
}
```

A Technical support and resources

- iDRAC Telemetry Workflow Examples
  1) GitHub - dell/iDRAC-Telemetry-Reference-Tools: Reference toolset for PowerEdge telemetry metric collection and integration with analytics and visualization solutions.

  2) https://github.com/dell/iDRAC-Telemetry-Scripting/

- Open source iDRAC REST API with Redfish Python and PowerShell examples.

  https://github.com/dell/iDRAC-Redfish-Scripting

- The iDRAC support home page provides access to product documents, technical white papers, how-to videos, and more.

  www.dell.com/support/idrac
- iDRAC User Guides and other manuals

  www.dell.com/idracmanuals

- Dell Technical Support
  Dell.com/support