DELL Technologies

Whitepaper

How To Integrate iDRAC9 Telemetry Data Into The Splunk Platform

Abstract

Dell EMC PowerEdge Servers running iDRAC9 version 4.0 or higher with the Datacenter license can stream DMTF Redfish telemetry data. This information helps IT administrators better understand the inner workings of their server environment. Telemetry data, simply put, is a series of timestamped numbers that represent different data points about your server.

This technical paper explains the steps required to setup Splunk to consume iDRAC9 telemetry. We will assume that you have Splunk set up but, if you do not, we will provide an abbreviated setup to assist getting a Splunk environment set up.

There are multiple methods for streaming the DMTF Redfish telemetry from each server as well as for implementing the Splunk environment. This is not an exhaustive document for all the different options. Additionally, services required in this implementation guide could be deployed on standalone hardware, virtualized and/or containerized (i.e. Docker) depending upon specific customer environments and needs.

September 2020

Revisions

Date	Description
9/24/2020	Version 1.0

Acknowledgements

This paper was produced by the following:

Name	
Kim Kinahan	Dell Technologies
Michael Brown	Dell Technologies
Kevin Tolly	The Tolly Group
Matheus Vieira	Dell Technologies
Tim Pacl	Dell Technologies
Kyle Prins	Dell Technologies
Dean Jackson	Splunk Inc.
Addison Lawrence	Dell Technologies
Rafael MarreroTorres	Dell Technologies
Tanuj Arcot	Dell Technologies
Chris Stahl	Dell Technologies

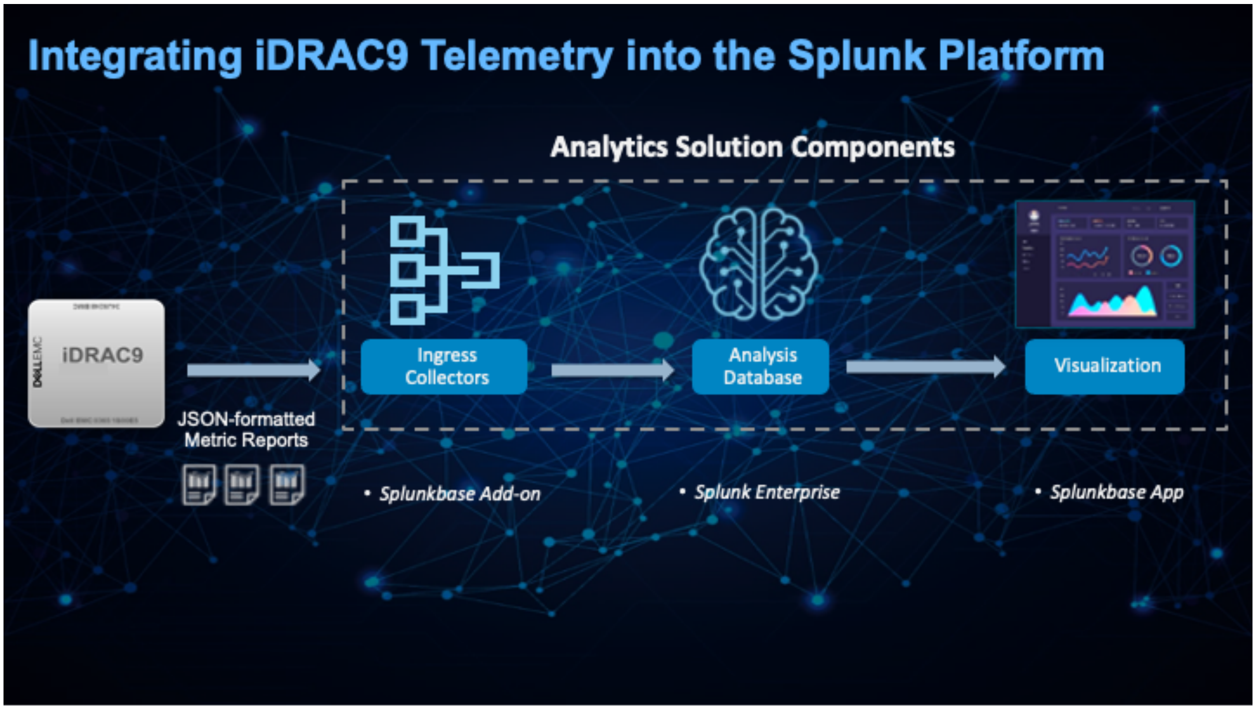
Table of Contents

Overview

The promise of "big data" can only be realized when that data is captured, stored and analyzed.

Dell EMC PowerEdge Servers provide a stream of telemetry data via iDRAC9 4.x in conjunction with a Datacenter license.[1] The Splunk platform can be used both to store that data from servers as well as to analyze that data visually. Telemetry streaming is more scalable and efficient than prior methods. For details concerning the efficiency, you can see the Tolly report on this topic.[2]

While iDRAC9 and Splunk are the only two systems needed for the end-to-end solution, there are multiple steps involved in bringing massive volumes of granular data to life. The functions span these components (and are discussed below): Data Generation (iDRAC9), Ingress Collectors, Analysis Database, and Visualization. These are shown in the following figure:


Integrating iDRAC9 Telemetry into the Splunk Platform
Analytics Solution Components
iDRAC9
JSON-formatted Metric Reports
Ingress Collectors
Analysis Database
Visualization
• Splunkbase Add-on
• Splunk Enterprise
• Splunkbase App

[1] iDRAC9 with v4.0 or later firmware and the Datacenter license are minimum server requirements for building the system described in this document.
[2] See the "Telemetry" tab under the Resources/White Papers section at www.dell.com/support/idrac

iDRAC9 Telemetry Overview

The iDRAC9 supports the standard DMTF Redfish “Telemetry” interface. You can read more information about this in the iDRAC Telemetry whitepaper. [3] This document, however, will guide you through a basic setup, specific for Splunk, that integrates the full power of iDRAC9 telemetry.

The telemetry interface in iDRAC9 will build “Metric Reports.” These reports are JSON documents that are easy to consume programmatically, but are also human-readable text. The main idea of the reports is that they have data similar in nature to the table below:

Metric Name	Device	Timestamp	Value
TemperatureReading	System Inlet Temp	2020-08-27 08:50:01	22
TemperatureReading	System Inlet Temp	2020-08-27 08:51:01	24
TemperatureReading	System Inlet Temp	2020-08-27 08:52:01	26
TemperatureReading	System Inlet Temp	2020-08-27 08:53:01	27

As one can see by looking at the sample data, it is easy to build things like graphs or other visualizations and do things like trend analysis or even more advanced things like predictive analytics.

Splunk Platform

Splunk is a commercial solution that can gather and store lots of data of different types: unstructured text logs as well as “metrics.”[4] Splunk then can perform search, analysis, even going up to advanced predictive alerting on that stored data in real time. When you set up Splunk with “Metric” collection, more advanced analytics and visualization options are available. It is not the intent of this paper to provide full and complete Splunk setup and configuration, however, we will go over a brief setup for a containerized Splunk environment just for completeness. We will show how to “ingest” the iDRAC9 Telemetry data as well as the iDRAC9 logs and alerts, so you have both the unstructured text and the metric data available.

In this paper, we will set up a Splunk add-on from Splunkbase to handle Telemetry ingest as well as the HTTP Event Collector (HEC) to handle logs and alerts. We will also go over setup for Splunk visualization and “dashboard” type additions that are also posted on Splunkbase. References below for all the utilities that we will be using in this doc.

Splunk: “Setting up Splunk in a container”:
https://docs.splunk.com/Documentation/Splunk/8.0.5/Installation/DeployandrunSplunkEnterpriseinsideDockercontainers

Additional, supplemental information on setting up Splunk:
https://hub.docker.com/r/splunk/splunk/

https://www.splunk.com/en_us/blog/it/an-insider-s-guide-to-splunk-on-containers-and-kubernetes.html

Splunk: HTTP Event Collector setup:
https://docs.splunk.com/Documentation/Splunk/8.0.5/Data/UsetheHTTPEventCollector

Splunkbase –Redfish Telemetry App:
https://splunkbase.splunk.com/app/5228/

[3] See the “Telemetry” tab under the Resources/White Papers section at www.dell.com/support/idrac

[4] This add-on does not use Splunk “metrics.”

Splunkbase – Redfish Telemetry Dashboards:
https://splunkbase.splunk.com/app/5245/

iDRAC9 Setup

The following steps should be performed on each iDRAC9. Download the Dell telemetry utilities. These will simplify configuration and let you script installation across many iDRACs without requiring manual GUI setup. The utilities are hosted at GitHub at the following URL: https://github.com/dell/iDRAC-Telemetry-Scripting.

Step 1: Download the telemetry utilities:

```
$ wget https://github.com/dell/iDRAC-Telemetry-Scripting/archive/master.zip -O iDRAC-Telemetry-
Scripting-master.zip
$ unzip iDRAC-Telemetry-Scripting-master.zip
$ cd iDRAC-Telemetry-Scripting-master
```

Step 2: Enable Telemetry and Metric Reports. Note in the command below, replace **$target** with the IP address or DNS name of the iDRAC9, replace **$user** with an iDRAC9 username with administrator privileges, and replace **$password** with the specified user's password.

```
$ python3 ./ConfigurationScripts/EnableOrDisableAllTelemetryReports.py -ip $target -u $user -p
$password
INFO:root:Successfully pulled configuration attributes
INFO:root:iDRAC Telemetry is currently 'Enabled'.
INFO:root:Successfully 'Enabled' iDRAC Telemetry and all reports.
```

Step 3 (Optional, see note below): Enable Redfish Logs and Alerts: this step will enable Redfish alerting, turn on the ability for iDRAC9 to publish Lifecycle Logs and Alerts, and also set up IDRAC9 to forward all of these to your Splunk server. (If setting up Splunk from this doc, come back to this step after you have set up your Splunk instance.) Note in the command below, replace **$target** with the IP address or DNS name of the iDRAC9, replace **$use**r with an iDRAC9 username with administrator privileges, and replace **$password** with the specified user's password. Additionally, replace **$splunkserver** with the IP address or DNS name of your Splunk HTTP Event Collector instance.

```
$ python3 ./ConfigurationScripts/SubscriptionManagementREDFISH.py -ip $target -u $user -p $password
-c y -D https://$splunkserver/services/collector/raw -E Alert -V Event
- WARNING, checking current value for iDRAC attribute "IPMILan.1.AlertEnable"
- WARNING, current value for iDRAC attribute "IPMILan.1.AlertEnable" is set to Disabled, setting
value to Enabled
- PASS, PATCH command passed to set iDRAC attribute "IPMILan.1.AlertEnable" to enabled
- PASS, iDRAC attribute "IPMILan.1.AlertEnable" successfully set to Enabled
- PASS, POST command passed, status code 201 returned, subscription successfully set for
EventService
```

The indication to look for with the above command, is the bolded last line, "status code 201, subscription successfully set for EventService."

NOTE: The step to enable Redfish Logs and Alerts is not strictly necessary for pure iDRAC9 Telemetry processing in Splunk. This step will forward all iDRAC9 Lifecycle Logs to your Splunk instance. We find that most customers want to have all Lifecycle Logs forwarded to Splunk, so this step is recommended but optional if you only want Telemetry.

Infrastructure Setup – Splunk

Section 1:

This section will walk you through a basic Splunk container setup. You can skip this if you already have a Splunk environment setup, or you can use this to set up a separate test environment to understand this new data before bringing it into your production environment.

Splunk publishes a Docker container with a simple installation, the details are here: https://hub.docker.com/r/splunk/splunk/, and the below steps are taken from that document.

Install Docker: See your Linux OS documentation or Docker documentation for how to install Docker on your server.

Start Docker and enable on boot

```
$ systemctl start docker
$ systemctl enable docker
```

Step 1: download the Splunk Docker image:

```
$ docker pull splunk/splunk:latest
```

Step 2: run the Docker image (replace **<password>** below with your own unique password). Note that below you are accepting the Splunk licensing terms by specifying the –accept-license parameter.

```
$ docker run -d -p 8000:8000 -e "SPLUNK_START_ARGS=--accept-license" -e "SPLUNK_PASSWORD=<password>"
--name splunk splunk/splunk:latest
```

After the Docker image is running, you can access your new Splunk install at http://localhost:8000

Splunk Add-on For Redfish Telemetry Reports Configuration

Note to reader: This section is intended for clustered environments where one needs to load the app files for all servers. For standalone implementations, one can use the GUI to install the app.

Standalone

To install the app in a standalone environment

1. Click on **Apps** menu, click **Find More Apps**.
2. Search keywords "Redfish Telemetry"
3. Click **Install**.
4. A message will state Splunk needs to restart. Click **Restart**.

Clustered Environment

After your Splunk Docker container is up and running, we will install the Add-on For Redfish Telemetry Reports. This add-on will be responsible for running in the Splunk environment, contacting the individual iDRACs and streaming telemetry from iDRAC9 to import into Splunk as Metrics.

To install apps and add-ons from within Splunk Enterprise

1. Log into Splunk Enterprise (**Heavy Forwarder** server's web interface).
2. On the **Apps** menu, click **Manage Apps**.
3. Click **Install app from file**.
4. In the **Upload app** window, click **Choose File**.
5. Locate the tar.gz file you downloaded from Splunkbase[5], and then click **Open** or **Choose**.
6. Click **Upload**.
7. Click **Restart Splunk**, and then confirm that you want to restart.

To install apps and add-ons directly into Splunk Enterprise

1. On the **Heavy Forwarder** server, put the downloaded file in the **$SPLUNK_HOME/etc/apps** directory.
2. Untar and ungzip y-on, using a tool like tar -xvf (on *nix) or WinZip (on Windows).
3. Restart Splunk.

Perform either one of the following set of instructions, depending upon if you are using a standalone or clustered environment.

Search Head (Standalone environment)

1. Deploy the add-on to your Search Head server (Please refer to Heavy Forwarder installation steps.[6])
2. Once done, navigate to **$SPLUNK_HOME/etc/apps/TA-redfish-add-on-for-splunk/default**
3. Copy props.conf file over from default folder into **../local/**
    1. Create local directory if it doesn't exist.
4. Edit the **props.conf** file, under **local** directory, and add the following parameters under each one of the redfish stanzas:

[5] https://splunkbase.splunk.com/app/5100/

[6] https://docs.splunk.com/Documentation/Splunk/8.0.6/AddMcafeeCloud/InstallHWF

```
KV_MODE = none
AUTO_KV_JSON = false
```

5. Restart the Search Head.

Search Head (Clustered environment)

1. Download and deploy the add-on to your Search Head Deployer under the following location **$SPLUNK_HOME/etc/shcluster/apps**
2. Once done, navigate to **$SPLUNK_HOME/etc/shcluster/apps/TA-redfish-add-on-for-splunk/default**
3. Copy props.conf file over from default folder into **../local/**
    1. Create local directory if it doesn't exist.
4. Edit the **props.conf** file, under **local** directory, and add the following parameters under each one of the Redfish stanzas:

```
KV_MODE = none
AUTO_KV_JSON = false
```

5. Push the changes from the Deployer server to the Search Head Cluster members by running the following command:
    ```
    splunk apply shcluster-bundle -target  https://<search head cluster captain>:8089
    ```
6. This procedure will trigger a rolling restart of your cluster; once done, the changes are in effect.

To configure new inputs from Redfish add-on for Splunk:

1. Once installed, open **Redfish add-on for Splunk** from Splunk UI
2. Navigate through **Configuration** tab
3. On **Account** tab, click the **Add** button
4. Add the credentials with access to Redfish Telemetry reports on client server
5. Once added the account, switch back to **Inputs** tab and click **Create New Input**; the following options need to be filled in the form:

Name: Specify input name
Interval: Time interval in secs for reports to be pulled from client server
Index: Where the events are going to be stored in Splunk
Global Account: The account added in the previous step
Hostname: Client host name or IP address
Chassis Collection Options: Default options are Overview, Power, Thermal. You can either add to them or remove them.
System Collection Options: Default options are Overview, Processors, Ethernet, Memory, Storage, Storage Subsystem.

6. Save the form, and the logs should start flowing into the selected index.

Note that the user may need to create an "Index" if one does not currently exist. Data Type is "Events."

Additional References

Tolly iDRAC9 Report

This Tolly-Dell report from February 2020 validates both the number of data points generated by telemetry streaming (compared with polling) along with quantifying the network efficiency of telemetry streaming over traditional polling. [7]

iDRAC9 Webinar

Join the discussion with Rick Hall (Systems Management Product Planning and Strategy), Doug Iler (iDRAC Product Manager), Michael E. Brown (Distinguished Engineer), and Kevin Tolly (3rd Party Testing & Validation) as they cover the new iDRAC9 v4.x Telemetry Streaming feature. In this 30-minute webinar, an overview of Telemetry Streaming, use cases, description of the feature validation testing and engineering perspective are covered. February 2020. See the link below:

https://www.brighttalk.com/webcast/13935/388223/idrac9-v4-0-telemetry-streaming

Transform Datacenter Analytics with iDRAC9 Telemetry Streaming

This 5-page tech note discusses the Telemetry Streaming feature part of the iDRAC9 Datacenter license. The Telemetry feature provides high-performance streaming of over 180 unique server and peripheral metrics with our industry-leading agent-free architecture.

https://www.dellemc.com/resources/en-us/asset/white-papers/products/software/direct-from-development-datacenter-telemetry.pdf

Dell iDRAC9 Telemetry Performance Reports

This technical paper details the Performance Reports and how to use them for monitoring and analyzing PowerEdge server utilization.[8]

Tolly iDRAC9 Splunk Use Case Report

This Tolly-Dell report from October 2020 demonstrates a use case with iDRAC9 telemetry integrated with Splunk and provides example visualizations in some areas key to delivering superior user experience. [9]

Unofficial Telemetry Streaming "How To" Blog by Dell's Jonas Werner

Jonas is a Dell Cloud Architect. In this very informative blog post, it details the steps required to implement both telemetry streaming and visualization using open-source tools (rather than using Splunk.)

https://jonamiki.com/2020/05/26/telemetry-streaming-with-dell-emc-poweredge-14g-servers-python-influxdb-and-grafana/

[7] See the "Telemetry" tab under the Resources/White Papers section at www.dell.com/support/idrac
[8] See the "Telemetry" tab under the Resources/White Papers section at www.dell.com/support/idrac
[9] See the "Telemetry" tab under the Resources/White Papers section at www.dell.com/support/idrac