z/OS 3.1 IBM Education Assistant

Solution Name: z/OSMF RTD REST API support

Solution Element(s): z/OSMF REST Hub

July 2023

Agenda

- Trademarks
- Objectives
- Overview
- Usage & Invocation
- Interactions & Dependencies
- Upgrade & Coexistence Considerations
- Installation & Configuration
- Summary
- Appendix

Trademarks

- See url http://www.ibm.com/legal/copytrade.shtml for a list of trademarks.
- Additional Trademarks:
  - None.

Objectives

- Explain new function for z/OSMF RTD REST API support.

Overview

- Who (Audience)
  - z/OS system administrator and application developer

- What (Solution)
  - z/OSMF provides a standard HTTPs REST API that can be invoked by web client to get runtime diagnostic data returned by RTD API about a specified set of address spaces on a defined system.

- Wow (Benefit / Value, Need Addressed)
  - z/OS provides a REST endpoint available to any external product for getting z/OS anomaly diagnostic data for anomalous subsystem workloads. The endpoint is delivered via this RTD REST API and can be invoked for specified systems (members) within the same parallel sysplex.

Usage & Invocation

- The z/OSMF Runtime Diagnostics (RTD) REST services are an application programming interfaces (API) that are implemented through industry standard Representational State Transfer (REST) services. The RTD services provide a programming interface to get runtime diagnostic data about a specified set of address spaces on a defined system.
- For a successful request, this response takes the form of an HTTP 200 status code and a result set that is passed back to your program as a JSON object or Text/Plain.
- For an unsuccessful request, the server response consists of a non-OK HTTP response code and details of the error, which are provided in the form of a JSON object.

Usage & Invocation

- **HTTPs method and URI path**
  - GET: /zosmf/resthub/hzr/<version>/analyze/anomaly?[sysname=name]&[asname=name&asname=name…]
  - Where:
  - **<version>** is the URI path variable that identifies the version of the z/OS runtime diagnostic service. The following value is valid: v1.
  - **Sysname** is the name of an optional CHAR(8) input variable that specifies the name of the system to be analyzed by RTD. The name is left-justified and padded on the right with blanks.
  - **Asname** is the name of an optional CHAR(8) input variable that contains a list of address space names or a name prefix. The first character in the name must be an alphabetic character or a special character (the symbols #, @, and $). The remaining characters can be alphanumeric or special characters, blank spaces cannot be included in a name. This path supports a maximum of four address space names at a time.

- **Custom headers X-IBM-Data-Type**
  - This header indicates which response data type is returned. The following type are supported:
    - application/json
    - text/plain
    - Default: The JSON response data will be returned

Usage & Invocation

Example: The following request obtains the abnormal data in the system sys1:

GET: `/zosmf/resthub/hzr/v1/analyze/anomaly?sysname=sy1`

Header: `X-IBM-Data-Type:`*`application/json`*

Response: *`status code: 200`*

Usage & Invocation

Example response


```
{

  "HZR0200I RUNTIME DIAGNOSTICS RESULT   SUMMARY": "SUCCESS",

  "REQ": "8",

  "TARGET SYSTEM": "SY1",

   "HOME": "SY1",

   "TIME": "2023/01/31 02:27:48",

   "INTERVAL": " 60 MINUTES",

         "TYPES": [

      "JES2:1"

   ],

   "EVENTS": [

      {

         "EVENT": "1",

         "PRIORITY": "HIGH",

         "TYPE": "JES2",

         "MESAGE": "$HASP9250 RESOURCE TGS UTILIZATION IS AT 83.80%",

         "ERROR": "INDICATED RESOURCE IS ABOVE THE WARNING THRESHOLD.",

         "ACTION": "LOOK FOR MESSAGE $HASP050. FOLLOW THE MESSAGE DOCUMENTATION FOR THE INDICATED RESOURCE."
```

Interactions & Dependencies

- Software Dependencies
  - z/OS Runtime Diagnostic
- Hardware Dependencies
  - NONE
- Exploiters
  - IZAA

Upgrade & Coexistence Considerations

- To exploit this solution, all systems in the Plex must be at the new z/OS level:
- Yes
- List any toleration/coexistence APARs/PTFs.
- N/A
- List anything that doesn't work the same anymore.
- N/A
- Upgrade involves only those actions required to make the new system behave as the old one did.
- N/A
- Coexistence applies to lower level systems which coexist (share resources) with latest z/OS systems.
- N/A

Installation & Configuration

- List anything that a client needs to be aware of during installation and include **examples** where appropriate - clients appreciate these:
  - Are any APARs or PTFs needed for enablement? No additional APARs are needed for V3.1.
  - What jobs need to be run? N/A
  - What hardware configuration is required? N/A
  - What PARMLIB statements or members are needed? N/A
  - Are any other system programmer procedures required? N/A
  - Are there any planning considerations? N/A
  - Are any special web deliverables needed? N/A
  - Does installation change any system defaults? N/A

Summary

z/OSMF RTD REST API support has been explained.

Appendix

- Please refer to z/OSMF Configuration Guide and z/OSMF Programming Guide for more details about z/OSMF RTD REST API.