How to handle SDK or Swagger related issue

Last updated by | Binlu Wang | Apr 7, 2022 at 9:02 PM PDT

Contents

- Common cases
- Identify the issue
 - Powershell
 - .NET & Python
- Workaround

Common cases

Common issue related to SDK or Swagger can be concluded in these two cases:

- 1. Customer tried to create or update Pipeline / Dataset / ... in ADF with SDK. The operation failed while the same json payload works in UI. Or the operation succeeded, but in UI some properties are missing.
- 2. Customer tried to get Pipeline / Dataset / ... in ADF with SDK. The operation failed with exceptions like Unable to extract Or Unable to deserialize. Or the operation succeeded, but the returned value has some properties missing.

Identify the issue

The underlying method for SDK are HTTP requests. So if there are any difference in behavior between using SDK and using the browser/UI, please collect the HTTP request / response for both SDK and UI, and then compare them to find out which property is missing.

Powershell

For Azure powershell commands, adding a -debug tag could enable the logs for HTTP request and response easily.

Get-AzDataFactoryV2IntegrationRuntime -ResourceGroupName myrg -DataFactoryName mydf -Name myir -debug

.NET & Python

Reproduce the issue with .NET SDK and capture HTTP requests with Fiddler.

Workaround

The publishing process for fixing the SDK related issue usually takes a month or more. Thus we'd suggest customer to use direct REST API method as workarounds.

Here is a powershell example for getting a pipeline:

```
$Subscription = "dald7b9a-a759-41c8-bb73-093a1818e03a"
$ResourceGroup = "my-ng"
$DataFactory = "my-df"
$Token = Get-AzAccessToken
$Headers = @{Authorization = "Bearer $($Token.Token)"}
$PipelineName = "pipelinel"
$Uri = "https://management.azure.com/subscriptions/$($Subscription)/resourcegroups/$($ResourceGroup)/providers/Microsoft.DataFactory/factories/$($DataFactory)
Invoke-RestMethod -Method GET -Uri $Uri -Headers $Headers
```