Data plane and exploratory API call failures

Last updated by | Jackie Huang | Jan 4, 2022 at 12:24 AM PST

Data plane and exploratory API call failures

Wednesday, February 5, 2020
08:41 PM

Issue:

Calls to get data plane token or exploratory APIs (test connection, import schema, preview data) fail, typically from Madrid UX

Resolution:

Exploratory APIs are primarily designed for interactive authoring scenarios, allowing customers to test connections to their external resources, import schema, and preview data when creating new linked services and datasets. Since those APIs query customer resources, possibly through a self-hosted integration runtime, they frequently exceed the ARM control plane API response time limits causing customers to see 504 errors when going through ARM, and ADF introduced a data plane to avoid those limits, and the general flow is to first call control plane through ARM (getDataPlaneAccess API) to get a response that includes a data plane Url and an accessToken that lasts for 8 hours, and use that Url and token to call data plane APIs directly without going through ARM. There are several possible paths:

1. Control plane. Typical for customers using UX on preview version factories, and customers using SDKs or REST to call control plane exploratory APIs directly, and customers with unusual exposure control settings, and maybe Fairfax. These look like other ADF APIs, so you'll find them in ARM, in ADF RP ClusterResourceProviderService role, and in ADMS.
    a. If customers hit 504 gateway timeout, and logs show they went through control plane path and did not hit any errors before reaching the ARM limit (50 seconds):
        i. If using UI with preview api version, recommend upgrading factory to GA.
        ii. If using SDK/REST, first confirm using latest GA SDK or 2018-06-01 api-version with REST. Explain that those APIs are intended for interactive UI scenarios, and check whether they can accomplish their desired scenario some other way (typically by running pipelines with GetMetadata and Lookup activities). If they still want to switch to data plane exploratory API, then note that the data plane apis are not yet documented and still subject to change, however they can use f12 browser debugging when running our UI to discover the current protocol. If they are top customers and request engineering assistance, file IcM to ADF Activities and UX.
            1. Adobe is a notable example of a top customer with automated REST calls to exploratory APIs, and they are already getting PG engineering support to migrate to data plane, although as of April 2019 they have not completed that migration.
2. Data plane. Typical for customers using UX on GA factories. The getDataPlaneAccess API is typical through ARM and entirely processed in ADF. Subsequent calls to exploratory APIs use URL and accessToken (in a header) to call RP data plane. The data plane calls do not show up in ARM, show up in ADF ApiOperationEvent and AdfTraceEvent as RP calls to DataPlaneRPService role, and show up normally in ADMS. There are currently two variations:
    a. API management. Example url prefix: dpwestcentralus.svc.datafactory.azure.com/dataplane/. API management maps this to underlying regional RP endpoint with prefix RPWestcentralus.svc.datafactory.azure.com:4433.
    b. No API management: Example url prefix: RPWestcentralus.svc.datafactory.azure.com:4433. Depending on customer environment, they may need to open extra outbound port to make this work, which is why we're rolling out the API management approach. In the meantime, if customers blocked, file IcM requesting EC change to force Madrid to use control plane for exploratory API on impacted subscription(s).

Common Case Patterns:

1. Failed to resolve the data factory server name **{data-plane-endpoint}** or get response from server, please refer to troubleshooting doc and check network configuration. Contact support if the issue persists. Activity ID: **{activity-id}**
    a. First follow this guide Troubleshoot Azure Data Factory UX Issues to see whether it's client-side network connectivity issue (e.g., DNS resolution, firewall, etc.) If yes, please DO NOT involve ADF PG because it's not something that Microsoft can help. Please directly ask the customer to reach their own corporate network team or ISP for mitigation and RCA.
        a. Search the browser trace (.har files) you collected from the customer browser, if you see the data plane requests failed with error **"net::ERR_NAME_NOT_RESOLVED"**, this is customer side DNS resolution issue.
        b. Double confirm that there is no ADF log (indicating the request did not even go to ADF backend service) with this Kusto query: **ApiOperationEvent** | where env_cloud_role == "DataPlaneRPService" | where TraceCorrelationId == "**{activity-id}**"
    b. If you feel it's not likely to be client-side network issue, please use the **{activity-id}** to query ADF Kusto logs first to understand what's happening roughly.
        a. **ApiOperationEvent** | where env_cloud_role == "DataPlaneRPService" | where TraceCorrelationId == "**{activity-id}**"
            a. If the HTTP response status code (indicated by the **resultSignature** field) is **499**, it's usually ADMS query timeout and client browser cancelled the request after a certain timeout. Use query (b) below to double confirm.
        b. **AdfTraceEvent** | where env_cloud_role == "DataPlaneRPService" | where TraceCorrelationId == "**{activity-id}**"
    c. If there is not much useful information in ADF Kusto logs, use the **{activity-id}** to further query ADMS Kusto logs.
    d. If you still want to open incident (ICM) to PG team, please share all the investigation results you have collected in step (a), (b) and (c) above in the incident to so that PG team doesn't need to waste time doing the same investigation again.

Once in ADMS, as usual for API calls the ADF TraceCorrelationId is the ADMS ActivityId, and the most interesting Azuredmprod tables are CustomLogEvent (probably sufficient for most troubleshooting), Details, TraceVerbose, TraceEvent, Operations, BillingEvent.

The ASC API reports and Troubleshooter should already show the most relevant log entries for failures and timeouts.

Additional Information:

Icm References:

Keywords:

Authors: arthurw

Reviewers: vimals

Created with Microsoft OneNote 2016.

How good have you found this content?