

ARM2ARM 4-Hour Timeout_Mig Move

Last updated by | Konstantin Chalyy | Mar 2, 2023 at 9:21 AM PST

Tags

[cw.Mig-Move](#)[cw.TSG](#)[cw.Reviewed-03-2023](#)

MICROSOFT CONFIDENTIAL

All content in this article is Microsoft Confidential and this data is considered covered under the Non-Disclosure Agreement (NDA).

This data should not be shared outside of the Microsoft Global Support organization, unless otherwise specified.

For more information, see *Microsoft Privacy Statement* [☐](#).

Contents

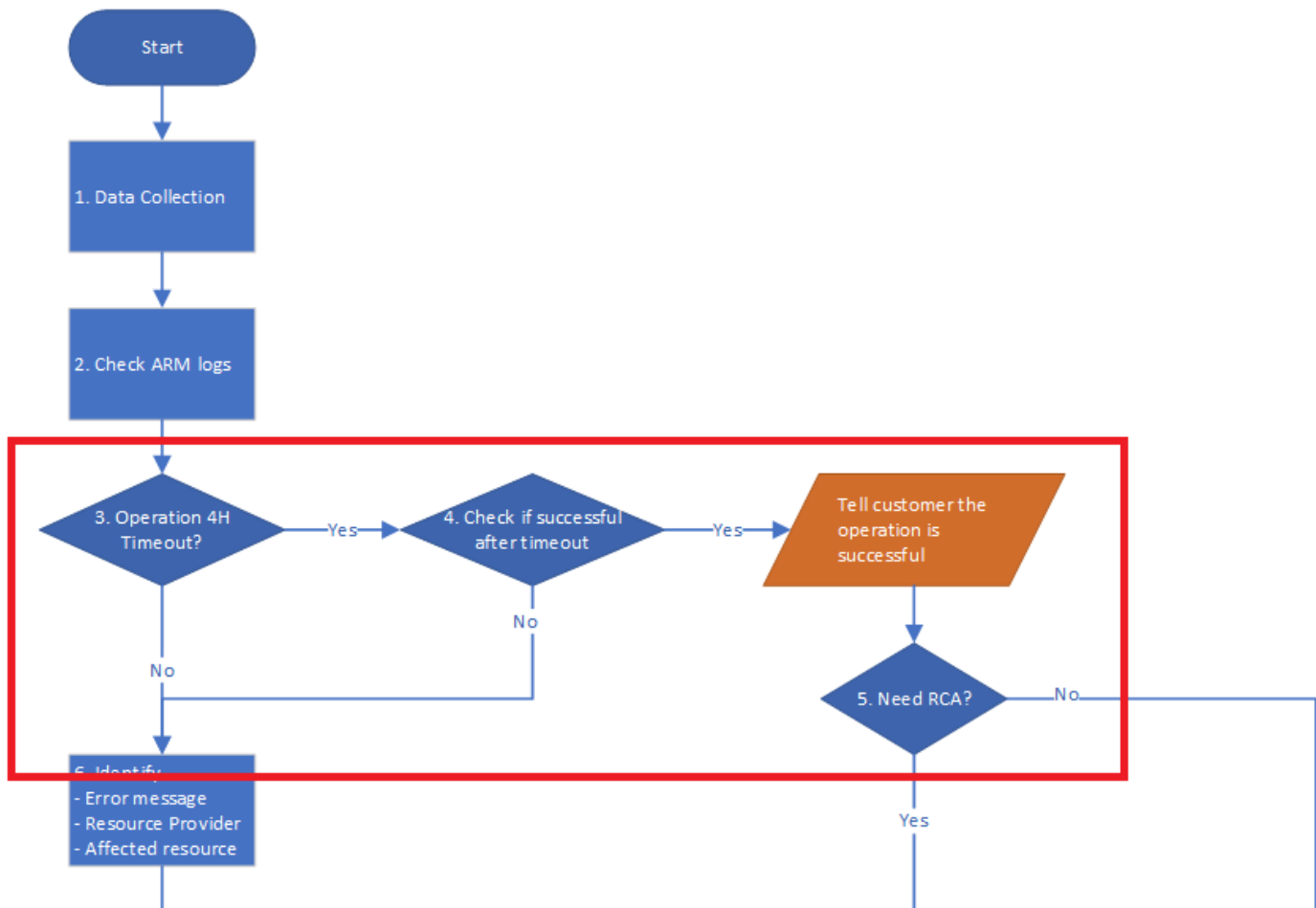
- [Introduction](#)
- [What is 4-hour timeout?](#)
- [Data Collection](#)
- [Troubleshooting](#)
 - [Mitigation:](#)
- [Need additional help or have feedback?](#)

Introduction

This document will explain how to troubleshoot a 4-hour timeout of ARM Move operation. This includes the following scenarios.

- Moving resources between resource groups in a single subscription.
- Moving resources between resource groups between subscriptions.

We will focus on the following part on [ARM2ARM move operation TSG](#):



Recently we don't have many case related to this 4-hour timeout issue, compared to 2019. ARM PG has added some improvement and the volume is decreasing at this point (2020.8). However, if you have the same 4-hour timeout issue, please take the cases carefully because there might be possibility that the consistency between resources breaks down.

What is 4-hour timeout?

The article [Frequently asked questions](#) can be provided to the customer as a simple summary of what took place.

When a user tries to move resources to another resource group, ARM handles the move operation and sends necessary API calls to each resource group. During this time, all the resources being moved are locked to keep consistency or avoid any change during the move.

However, the lock doesn't last forever. It has a timeout value of 4 hours (240 mins). After 4 hours have passed, the move operation will fail with a "timeout" message and all the locks for resources will be released.

You may see the following error:

```
{
  "statusCode": "Conflict",
  "statusMessage": {
    "error": {
      "code": "ResourceMoveTimedOut",
      "message": "Moving resources did not finish within allowed time '04:00:00'. Provisioning state of the resource groups will be rolled back."
    }
  }
}
```

Note: Error message is subject to change in the future update.

This lock behavior is described in the document: [Move resources to a new resource group or subscription](#) .

Both the source group and the target group are locked during the move operation. Write and delete operations are blocked on the resource groups until the move completes. This lock means you can't add, update, or delete resources in the resource groups. It doesn't mean the resources are frozen. For example, if you move a SQL Server and its database to a new resource group, an application that uses the database experiences no downtime. It can still read and write to the database. The lock can last for a maximum of four hours, but most moves complete in much less time.

Note: When a customer has a long-running move operation, they may be upset because they are unable to perform certain 'update' processes on the resources. Unfortunately, this is by-design. Please tell the customer that they need to wait for the move operation to complete or error out.

Data Collection

If you follow the general [ARM2ARM Move TSG](#), you should have the necessary information at **Data Collection** phase.

- Precise Start Timestamp
- Correlation Id

You can retrieve the activity/operation related by using **correlation id**. That's why you need this correlation id when you investigate issues in ARM.

Troubleshooting

If 4-hour timeout happens, you should check if there are any resources missing both a source and destination resource group. In some cases, the write operation on the destination fails and the link from ARM gets lost.

Even after a 4-hour timeout error message, the underlying operation may finish successfully. The current status of the move operation can be checked using the Kusto queries below.

1. Locate the "Correlation ID" that will be used for additional queries.

Correlation ID query:

```
// Find the CorrelationID
cluster('azcsupfollower.kusto.windows.net').database('ARMProd').EventServiceEntries
| where operationName contains "moveresources"
| where subscriptionId == "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
| where PreciseTimeStamp > ago(30d)
//| where PreciseTimeStamp >= datetime(2019-12-01T00:00:01.000000Z) and PreciseTimeStamp <= datetime(2019-12-
```

2. Check the current status of the move operation.

Move operation status query:

```
cluster('azcsupfollower.kusto.windows.net').database('ARMPProd').EventServiceEntries
| where PreciseTimeStamp between (datetime({starttime})..datetime({endtime}))
| where correlationId == '{correlationid}'
| project PreciseTimeStamp, ActivityId, correlationId, operationName, operationId,
resourceProvider, level, status, subStatus, httpRequest, properties, resourceUri,
armServiceRequestId, authorization, claims, RoleInstance, SourceNamespace
```

The query results may be similar to this:

| PreciseTimeStamp | ActivityId | correlationId | operationName | operationId | resourceProvider | level | status | subStatus | httpRequest |
|-----------------------------|------------|--------------------------------------|---|-------------|---------------------|-------|-----------|-----------|-------------------------------|
| 2019-07-29 13:53:59.5254533 | | | Microsoft.Resources/subscriptions/resourceGroups/moveResources/action | | Microsoft.Resources | 4 | Started | | ['clientRequestId':'f85fe4f-c |
| 2019-07-29 13:54:00.9437568 | | | Microsoft.Resources/subscriptions/resourceGroups/moveResources/action | | Microsoft.Resources | 4 | Accepted | Accepted | ['clientRequestId':'f85fe4f-c |
| 2019-07-29 13:55:32.7734936 | | | Microsoft.Resources/subscriptions/resourceGroups/moveResources/action | | Microsoft.Compute | 4 | Started | | |
| 2019-07-29 13:55:32.8324911 | | | Microsoft.Resources/subscriptions/resourceGroups/moveResources/action | | Microsoft.Compute | 4 | Accepted | | |
| 2019-07-29 13:55:48.3867858 | | | Microsoft.Resources/subscriptions/resourceGroups/moveResources/action | | Microsoft.Compute | 4 | Succeeded | | |
| 2019-07-29 13:55:48.5629695 | | | Microsoft.Compute/disks/write | | Microsoft.Compute | 4 | Started | | |
| 2019-07-29 13:55:49.3475567 | | | Microsoft.Compute/disks/write | | Microsoft.Compute | 4 | Succeeded | | |
| 2019-07-29 13:55:49.3477828 | | | Microsoft.Compute/disks/delete | | Microsoft.Compute | 4 | Started | | |
| 2019-07-29 13:55:51.0255963 | | | Microsoft.Compute/disks/delete | | Microsoft.Compute | 4 | Succeeded | | |
| 2019-07-29 14:05:52.6023809 | | | Microsoft.Compute/disks/delete | | Microsoft.Compute | 4 | Succeeded | | |
| 2019-07-29 14:05:52.7935622 | | | Microsoft.Compute/disks/write | | Microsoft.Compute | 4 | Succeeded | | |
| 2019-07-29 17:54:31.0063367 | | edc4c350-2219-4700-b0a3-12371760a00e | Microsoft.Resources/subscriptions/resourceGroups/moveResources/action | | Microsoft.Resources | 2 | Failed | | |

Example 4-hour timeout error:

```
{"statusCode":"Conflict","statusMessage":{"error":{"code":"ResourceMoveTimedOut","message":"Moving
resources did not finish within allowed time '04:00:00'. Provisioning state of the resource groups will be rolled
back."}}}
```

- Check which job is stuck. Basically, ARM creates necessary jobs and each job has responsibility to complete a specific task. In most cases, one of the jobs is taking a longer time than expected (over 4 hours). This is why the 4-hour timeout occurs. You can check the job operation with the following Kusto query.

Individual job status query:

```
cluster('azcsupfollower.kusto.windows.net').database('ARMPProd').JobTraces
| where PreciseTimeStamp between (datetime({starttime})..datetime({endtime}))
| where correlationId == '{correlationid}'
| project PreciseTimeStamp, Level, ActivityId, subscriptionId, correlationId, operationName,
jobPartition, jobId, message, exception, RowKey
```

- Identify which operation is failing. Even after the 4-hour timeout error has been observed, if each job has finished successfully and there's no missing resources in the destination resource group, the move operation should be successful. You can tell the customer that the operation finished after the 4-hour timeout.

However, in most cases, one of the jobs is failing to finish. You need to identify which operation is failing. For example, you may see the following logs.

```
'2' out of '2' Resources got moved successfully.
Some linked notification actions have not completed for moving resource Id 'Microsoft.Storage/storageAccounts/
Some linked notification actions have not completed for moving resource Id 'Microsoft.Storage/storageAccounts/
Frontdoor job completed with status: 'Postponed', message: 'Linked notifications are not complete for '2' out
```

5. Contact the service owner of the failing operation. In this example, the job is calling an API of Microsoft.Storage to check the current state. You can check which API is called by the following Kusto query:

Job and API relationship Query:

```
cluster('azcsupfollower.kusto.windows.net').database('ARMPProd').HttpOutgoingRequests
| where PreciseTimeStamp between (datetime({starttime})..datetime({endtime}))
| where correlationId == '{correlationid}'
| project PreciseTimeStamp, Level, ActivityId, TaskName, subscriptionId, correlationId, operationName,
  httpMethod, hostName, targetUri, httpStatusCode, errorCode, errorMessage, durationInMilliseconds,
  contentLength, referer
```

You need to engage the service owner of this resource provider (in this case, Microsoft.Storage).

Mitigation:

If you have already explained about the 4 hour timeout, per the [Frequently asked questions](#) article, and customer is looking for root cause, proceed to file an IcM using escalation options in ASC

Need additional help or have feedback?

| <i>To engage the Azure Migration and Move SMEs...</i> | <i>To provide feedback on this page...</i> | <i>To provide kudos on this page...</i> |
|---|---|--|
| <p>Please reach out to Azure Migration and Move SMEs ☑</p> <p>Make sure to use the Ava process for faster assistance.</p> | <p>Use the Azure Migration and Move Feedback form to submit detailed feedback on improvements or new content ideas for Azure Migration and Move.</p> <p>Please note the link to the page is required when submitting feedback on existing pages!</p> <p>If it is a new content idea, please put N/A in the Wiki Page Link.</p> | <p>Use the Azure Migration and Move Kudos form to submit kudos on the page. Kudos will help us improve our wiki content overall!</p> <p>Please note the link to the page is required when submitting kudos!</p> |