

ADF access to Public AWS environment (untrusted endpoints without having access key)

Last updated by | Anil K B | Sep 19, 2022 at 1:24 AM PDT

Contents

- [Issue occurrence scenarios:](#)
- [How to access untrusted endpoints where cx did not have ...](#)



Issue occurrence scenarios:

1. When cx wants to connect ADF with S3 linked service to a public Amazon site to get some resources from that endpoint, and cx is not able to do it because cx does not know the Access Key Id.

How to access untrusted endpoints where cx did not have the access key, because it is public

1. Authentication type in S3 linked service, use **Temporary Security Credentials** instead of Access Key:

New linked service

 Amazon S3 [Learn more](#) 


Name *

Description

Connect via integration runtime * ⓘ

 AutoResolveIntegrationRuntime  

Authentication type

 Access key

Access key

Temporary security credentials

GetSessionToken—temporary credentials for users in untrusted environments

The `GetSessionToken` API operation returns a set of temporary security credentials to an existing IAM user. This is useful for providing enhanced security, such as allowing AWS requests only when MFA is enabled for the IAM user. Because the credentials are temporary, they provide enhanced security when you have an IAM user who accesses your resources through a less secure environment. Examples of less secure environments include a mobile device or web browser. For more information, see [Requesting temporary security credentials](#) or [GetSessionToken](#) in the *AWS Security Token Service API Reference*.



2. When selecting **Temporary security credentials**, it asks for Session token:

New linked service
Amazon S3 [Learn more](#)

Name *

AmazonS31

Description

Connect via integration runtime * ⓘ

AutoResolveIntegrationRuntime

Authentication type

Temporary security credentials

Access key ID Azure Key Vault

Access key ID *

Secret access key Azure Key Vault

Secret access key *

Session token Azure Key Vault

Session token *

ⓘ AWS temporary credential expires between 15 minutes to 36 hours based on settings. Make sure your credential is valid when activity executes. [Learn more](#)

3. Please refer the AWS document which describes more on session token - [Requesting temporary security credentials - AWS Identity and Access Management \(amazon.com\)](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_temp.html), https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_temp.html ⓘ
4. The session token MUST be copied from the **AWS (Amazon) side**. It seems that this token is visible in the **User Admin Profile** or **User Profile** in AWS side. Accordingly, to the customer this token is generated automatically. And sometimes, it seems that this token is aged the first time the customer retrieves it from his AWS profile. The token MUST be fresh, otherwise it will fail the connection. Accordingly, to the customer he had to wait a few days until he gets the new token. This is what our documentation says- <https://learn.microsoft.com/en-us/azure/data-factory/connector-amazon-simple-storage-service?tabs=data-factory> ⓘ.

sessionToken Applicable when using temporary security credentials [authentication](#). [Learn how to request temporary security credentials](#) [from AWS](#). No

[Note AWS temporary credential expires between 15 minutes to 36 hours based on settings.](#)
Make sure your credential is valid when activity executes, especially for operationalized workload - for example, you can refresh it periodically and store it in Azure Key Vault.
Mark this field as a SecureString to store it securely, or reference a secret stored in Azure Key Vault.

Scenario like this would take lot of time to refer multiple documents b/w Microsoft and AWS and would eventually lead more time to resolve the cx issue. This TSG should help CSS team in solving the case swiftly in the scenarios cx wants to establish a connection from ADF to Public AWS environment (untrusted endpoints without having access key).

How good have you found this content?

