Configure AWS Per-User Authentication

Contents:

For Trifacta Wrangler Enterprise, you can configure AWS authentication on a per-user basis, using temporary credentials for superior security.

Enable

The following parameters must be set:

Property	Description
`"aws.readFromConfigurationService": false,`	Set this value to `false` for Trifacta Wrangler Enterprise, which prevents the product from retrieving AWS-related configuration information from the incorrect source.
`"aws.mode": "user",`	Each user can specify credentials.

To authenticate to AWS services from the Trifacta platform using an IAM role:

Property	Description

`"aws.ec2InstanceRoleForAssumeRole": true,`	• If `true`, then all users use the EC2 instance role for authenticating to the AWS STS service for their temporary credentials.
	
	NOTE: You must ensure that the role provides adequate access to STS. Details are below.
	
	Tip: This method is recommended.
	
	• If `false`, then a system-wide set of AWS key/secret credentials must be inserted into platform configuration in the Admin Settings page as the master set of credentials to access STS for temporary credentials:
	
	Properties to set:
	
	`"aws.s3.key"`
	`"aws.s3.secret"`
	
	NOTE: After specifying the above key/secret combination, you can skip to the User Access section below.

Configure Per-User Authentication using IAM Role

Please complete the following general steps.

Steps:

1. Instance role: Create an IAM role and link it to the EC2 instance where the Trifacta node is hosted. Include the following IAM policy:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "sts:AssumeRole",
            "Resource": "arn:aws:iam::*:role/*"
        }
    ]
}
```

2. User role: Create another IAM role and provides required access to the S3 buckets. Example:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "MyBucketAndObjectPermissions",
            "Action": [
                "s3:GetBucketLocation",
                "s3:ListBucket",
                "s3:DeleteObject",
                "s3:GetObject",
                "s3:PutObject"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:s3:::<my_s3_bucket>",
                "arn:aws:s3:::<my_s3_bucket>/*"
            ]
        },
        {
            "Sid": "TrifactaPublicDatasets",
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::trifacta-public-datasets/*",
                "arn:aws:s3:::trifacta-public-datasets"
            ]
        }
    ]
}
```

where:
`<my_s3_bucket>` is the name of your bucket.

3. Under the user role definition, edit the Trust relationship. Add the instance role to Principal:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
        "arn:aws:iam::       {awsAccountId}:role/{instanceRole}"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

   a. For more information, see *Insert Trust Relationship in AWS IAM Role*.
   b. For more granular control over the Trust relationship, see *https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_elements_principal.html*.
4. AWS Glue: If you are integrating with AWS Glue, additional permissions must be set. For more information, see *Enable AWS Glue Access*.
5. Log in the Trifacta platform as a Trifacta admin.
6. Click the link to specify storage settings. Populate the values for:
   a. IAM role
   b. Role ARN
   c. S3 Bucket Name
7. Save your changes.

User Access

After per-user authentication has been enabled, each user must provide or be provided the credentials and S3 bucket to use. Users can insert a default S3 bucket and credentials to use in their profiles. See *Configure Your Access to S3*.