CommunityWFM

Integration Specification AWS (Amazon Web Services)

For CommunityWFM Software Version 4.4+/5.0+

March 2024

3400 Waterview Parkway, Suite 325
Richardson, Texas 75080

phone 877-668-6870
web CommunityWFM.com

Table of Contents

About this Document

The objective of this document is to outline the method and details of the CommunityWFM integration to AWS (Amazon Web Services). The document assumes that the reader has some basic understanding of AWS including IAM users and policy management.

WFMSG Community and AWS Integration

CommunityWFM integrates with AWS through its programming API that allows CommunityWFM the ability to retrieve the required real-time data. AWS will meet data collection requirements to fully integrate the CommunityWFM application with the AWS platform. The IAM user in AWS that CommunityWFM will use to retrieve data must have the following permissions set with the appropriate information.

IAM policy needs to be updated to match the below Actions. Resource should be set to the proper setting for the AWS connection.

```
{
    "Version": "2012-10-17",
    "Statement":
    [
      {
        "Effect": "Allow",
        "Action":
         [
          "kinesis:GetShardIterator",
          "kinesis:GetRecords",
          "kinesis:DescribeStream",
          "connect:DescribeInstance",
          "connect:ListInstances",
          "connect:ListUsers",
          "connect:ListQueues",
          "connect:GetMetricData",
          "connect:GetCurrentMetricData",
          "connect:DescribeUser"
        ],
        "Resource":
         [
                "arn:aws:kinesis:us-east-1 :############:stream/AgentEventKinesisStreamForWFMSG",
                "arn:aws:connect:us-east-1:############:instance/ ********-****-****-****-************/*"
         ]
      }
    ]
}
```

Each component of data collection and the required database and tables are described below.

Historical Data Collection

CommunityWFM collects historical contact volume data from AWS by using the GetMetricData and GetCurrentMetricData in the API. The CommunityWFM Historical Data Collection Service (a .NET Windows service) executes the call for each defined data collection point every 15 minutes and loads the results into vendor-neutral tables inside the CommunityWFM product database.

Below is the configuration information needed for setting up the historical collection from AWS.
Note: Proxy information is only necessary if a proxy server is being used, otherwise leave blank.

Historical contact volume data collection

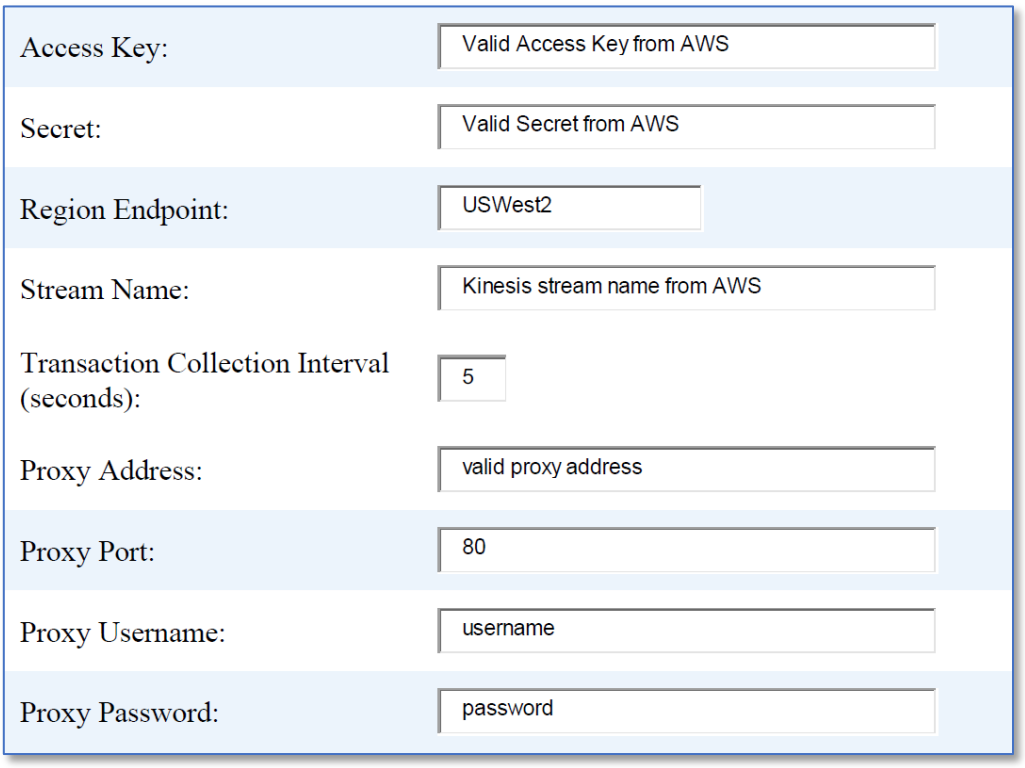
Access Key:	Valid Access Key from AWS
Secret:	Valid Secret from AWS
Assume Role ARN:	Valid ARN to Assume Role from AWS
Region Endpoint:	USWest2
Instance ID:	Valid Instance ID from AWS
Proxy Address:	valid proxy address
Proxy Port:	80
Proxy Username:	username
Proxy Password:	password

Agent State Transaction Data Collection

CommunityWFM collects agent state transactions from AWS to compare against schedule intervals to provide agent schedule adherence reporting. The CommunityWFM Adherence Collection Service (a .NET Windows service) executes the query on a user-defined interval (typically between 5 and 30 seconds) and loads the results into vendor-neutral tables inside the CommunityWFM product database.

To retrieve data from AWS, an AWS Kinesis stream must be configured in AWS to provide the real-time data feed. Proper permissions must be set via the IAM user policy.

Below is the required information needed to be able to receive that real-time data stream.
Note: Proxy information is only necessary if a proxy server is being used, otherwise leave blank.



Importing Configuration Data

Community supports the ability to import agents directly into CommunityWFM from the AWS instance via the API calls ListUser and DescribeUser. Permissions must be properly set for the IAM user.

Steps to create a User to Assume Role

Create a Policy

1. Name it: WFMSGConnectPolicy
2. Edit the policy to match this:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "connect:DescribeInstance",
                "connect:ListQueues",
                "connect:GetMetricData",
"connect:GetCurrentMetricData",
                "connect:ListInstances",
                "connect:ListUsers".
                "kinesis:GetShardIterator",
                "connect:DescribeUser",
                "kinesis:GetRecords",
                "kinesis:DescribeStream"
            ],
            "Resource": "*"
        }
    ]
}
```

3. Copy the ARN (in this example it is: arn:aws:iam::123456:policy/WFMSGConnectPolicy)

Create a Role

1. For type choose AWS Service
2. For use case choose S3 (we will change this later in the process)
3. Select S3 at the bottom and click next for Permissions
4. For name use "WFMSGConnectAPIRole"
5. Attach the policy we created above called "WFMSGConnectPolicy"
6. Copy the ARN (in this example it is: arn:aws:iam::123456:role/WFMSGConnectAPIRole)

Create a Policy

Name the policy "WFMSGConnectAssumePolicy"

Edit the policy to match this: (replace the example ARN below with the one from the the role just created)

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "VisualEditor0",
         "Effect": "Allow",
         "Action": "sts:AssumeRole",
         "Resource": "arn:aws:iam::123456:role/WFMSGConnectAPIRole"
      }
   ]
}
```

Create a User

1. Create a user with access type of "Programmatic access": (in this example we are using "WFMSGConnectAssumeUser" as the user name.
2. Generate an access key and record both the access key id and the access key secret
3. Assign that user the policy "WFMSGConnectAssumePolicy"
4. Copy the ARN (in this example it is: arn:aws:iam::123456:user/WFMSGConnectAssumeUser)

Edit a Role

1. Edit role "WFMSGConnectAPIRole"
2. Edit the Trust Relationship to match the: (replace the example ARN below with the one from the the user just created)

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "connect.amazonaws.com",
        "AWS": "arn:aws:iam::123456:user/WFMSGConnectAssumeUser"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```