



Implementation guide for Azure Active Directory for 21Vianet

<https://github.com/iamSmally/ABAP-SDK-for-Azure>

Author: iamSmally

Modified from: Implementation guide for Azure Active Directory

Version: 1.0

Contents

- [1. What is Azure Active Directory?](#)
- [2. Prerequisites](#)
- [3. How to setup Azure Active Directory in Azure?](#)
- [4. Generate keys for your application](#)
- [5. Steps to use AAD authentication from SAP using ABAP SDK for Azure](#)
- [5.1 Creation of RFC destination to Azure Active Directory](#)
- [5.2 Configuration](#)
 - [ZREST_CONFIG](#)
 - [ZREST_CONF_MISC](#)
 - [ZADF_CONFIG](#)
- [6. ABAP SDK Monitor](#)
- [7. Auto re-processing of failed messages](#)

What is Azure Active Directory?

Azure Active Directory (Azure AD) provides an easy way for businesses to manage identity and access, both in the cloud and on-premises. Your users can use the same work or school account for single sign-on to any cloud and on-premises web application. Users can use their favorite devices, including iOS, Mac OS X, Android, and Windows. An Organization can protect sensitive data and applications both on-premises and in the cloud with integrated multi-factor authentication ensuring secure local and remote access. Azure AD extends your on-premises directories so that information workers can use a single organizational account to securely and consistently access their corporate resources. Azure AD also offers comprehensive reports,

analytics, and self-service capabilities to reduce costs and enhance security. The Azure AD SLA ensures that your business always runs smoothly and can be scaled to enterprise levels.

For more details on Azure Active directory, visit [Microsoft Azure Active Directory](#)

Prerequisites

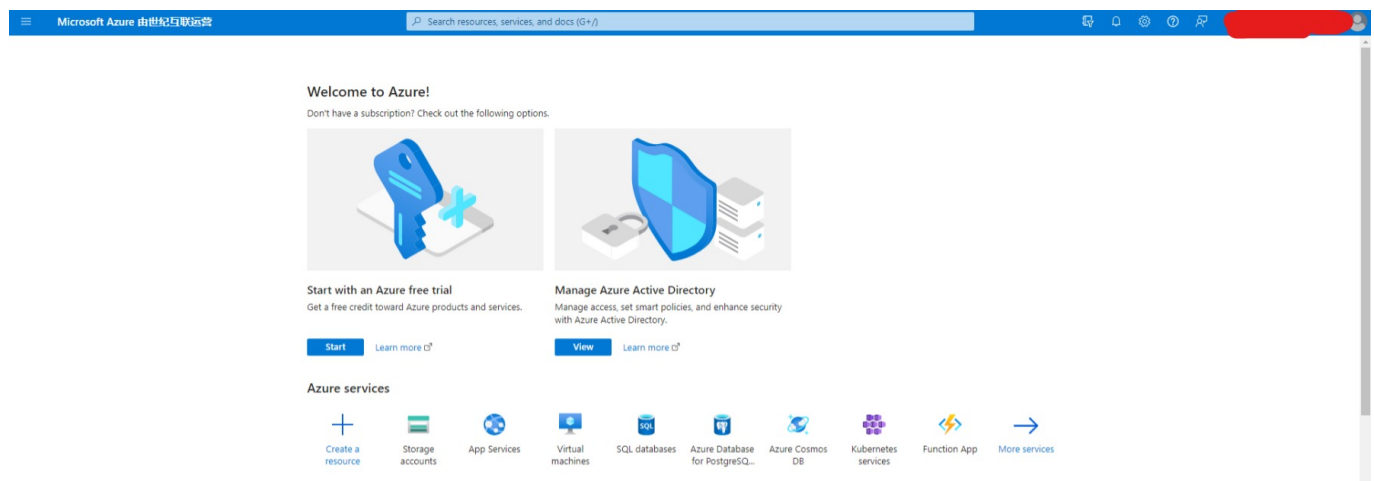
Make sure you have installed ABAP SDK for Azure in your SAP system. Refer document 'ABAP SDK for Azure – GitHub' for more details.

Visit <https://github.com/iamSmally/ABAP-SDK-for-Azure>

How to setup Azure Active Directory in Azure?

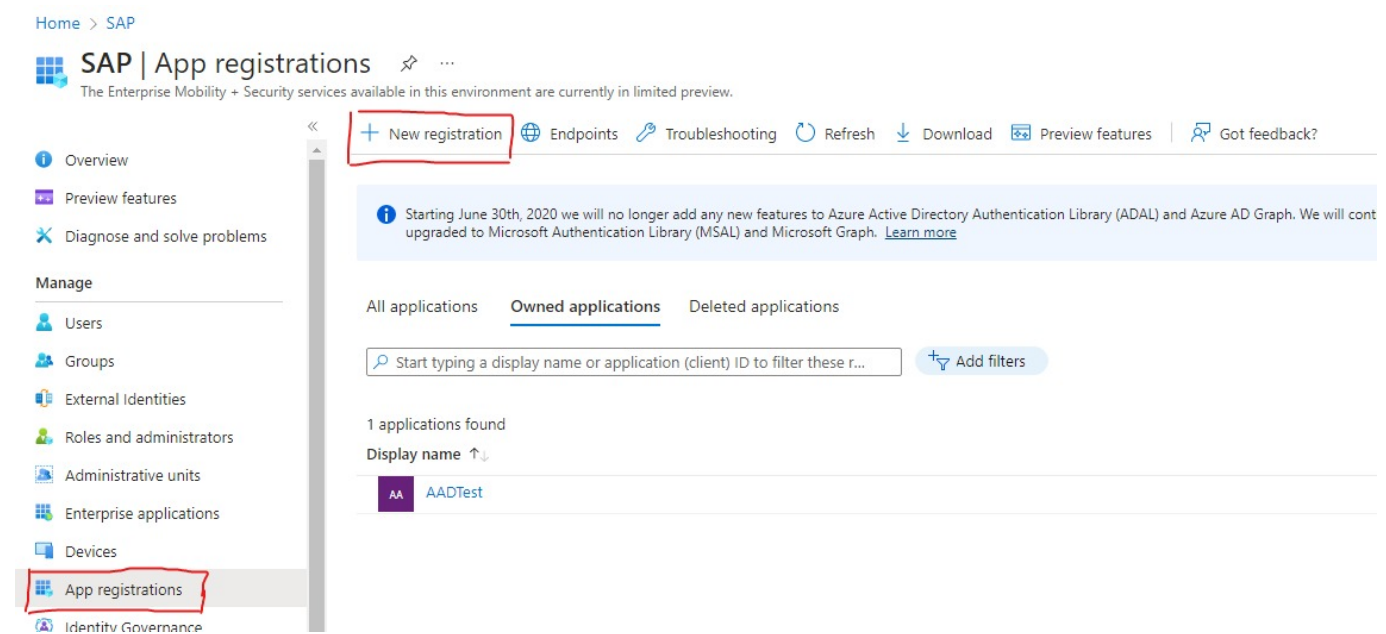
Login to [Microsoft Azure portal - managed by 21Vianet](#)

Note: If you do not have an account already, please create a new Azure account. You can start free. Once you are logged into portal, go to all services and search for Azure Active Directory and Select "Azure Active Directory" as shown below.

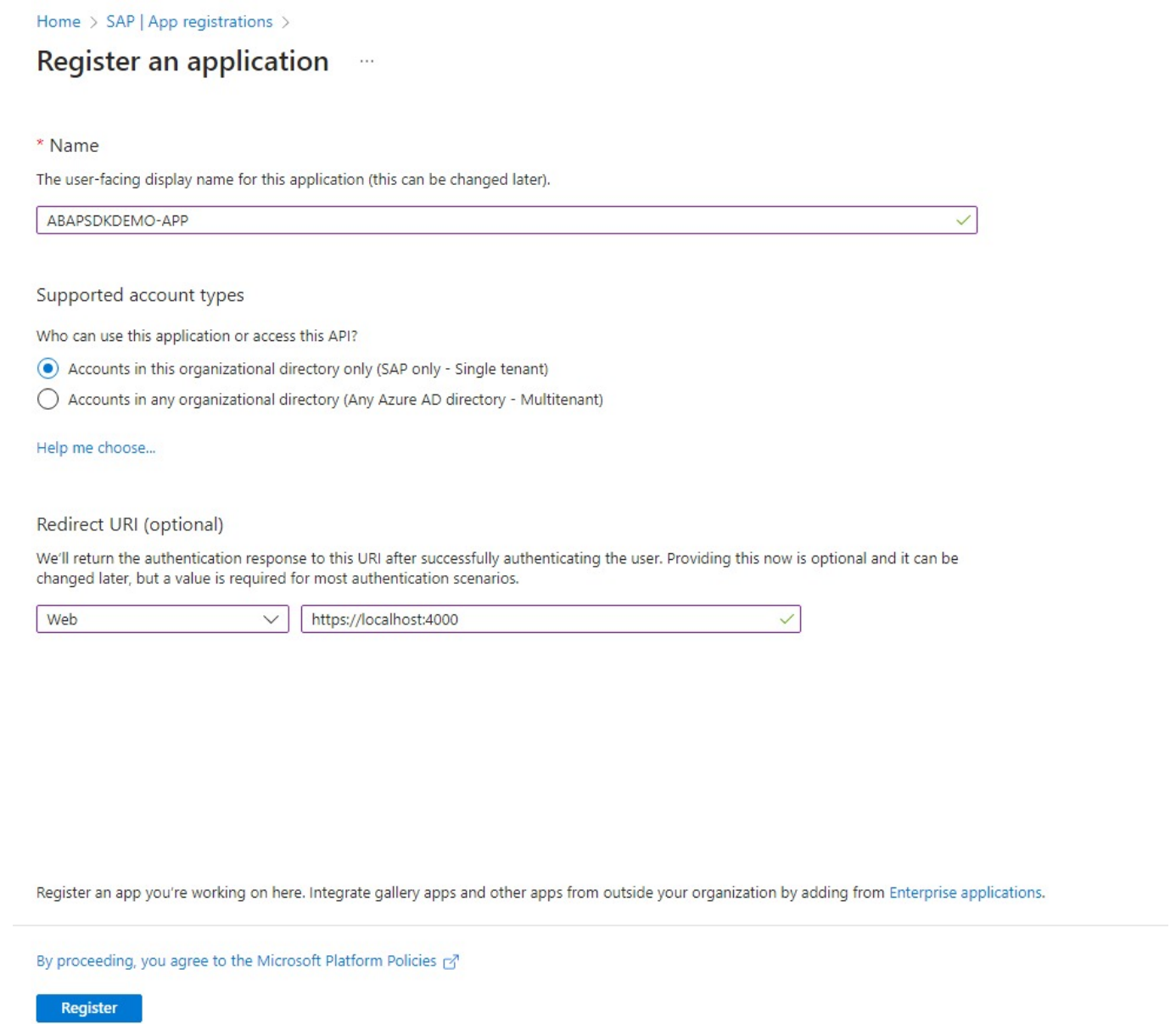


Create a new tenant for your organization in case it hasn't been created. <https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-access-create-new-tenant>

Click on 'App Registrations' on left side menu as shown below and click the button 'New Registration'



Specify details of your Application and press 'Register' button.



Application is created successfully.

Home > SAP

SAP | App registrations

The Enterprise Mobility + Security services available in this environment are currently in limited preview.

Overview
Preview features
Diagnose and solve problems

Manage

- Users
- Groups
- External Identities
- Roles and administrators
- Administrative units
- Enterprise applications
- Devices
- App registrations**
- Identity Governance
- Application proxy
- Custom security attributes (Preview)

+ New registration Endpoints Troubleshooting Refresh

Starting June 30th, 2020 we will no longer add any new features to Azure Act upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. [Learn more](#)

All applications **Owned applications** Deleted applications

Start typing a display name or application (client) ID to filter these r...

2 applications found

Display name ↑↓

AA	AADTest
AB	ABAPSDKDEMO-APP

Generate keys for your application

1. Once your application is created, go to your application by clicking on it. Copy the application id which will be required in the implementation of code in ABAP SDK. This application id is client Id. Click on 'Certificates & secrets',

Home > SAP | App registrations >

ABAPSDKDEMO-APP

Search << Delete Endpoints Preview features

Got a second? We would love your feedback on Microsoft identity platform (previous)

Essentials

Display name : [ABAPSDKDEMO-APP](#)

Application (client) ID [REDACTED]

Object ID [REDACTED]

Directory (tenant) ID [REDACTED]

Supported account types : [My organization only](#)

Starting June 30th, 2020 we will no longer add any new features to Azure Active D be upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. [Le](#)


[Get Started](#) Documentation

Navigation

- Overview
- Quickstart
- Integration assistant
- Manage
 - Branding & properties
 - Authentication
 - Certificates & secrets**
 - Token configuration
 - API permissions
 - Expose an API
 - App roles
 - Owners
 - Roles and administrators
 - Manifest
- Support + Troubleshooting
 - Troubleshooting
 - New support request


Bui


The Microsoft identity platform is an a



2. Then click on 'New client secret' button as shown in the below screen. Then press 'Add' button on the popup dialog.


Home > SAP | App registrations > ABAPSDKDEMO-APP

 ABAPSDKDEMO-APP | Certificates & secrets

 ...

Search

«

 Got feedback?

Overview

Quickstart

Integration assistant

Manage

Branding & properties

Authentication

Certificates & secrets

Token configuration

API permissions

Expose an API

App roles

Owners

Roles and administrators


Manifest

Support + Troubleshooting

Troubleshooting

New support request

Credentials enable confidential applications to identify themselves to the authentication scheme). For a higher level of assurance, we recommend using a certificate (if available).


 Application registration certificates, secrets and federated credentials can be managed in this page.

Certificates (0)

Client secrets (0)

Federated credentials (0)

A secret string that the application uses to prove its identity when requesting tokens from the identity provider.

 New client secret

Description	Expires	Value
No client secrets have been created for this application.		

Copy this key and it will be used in ABAP SDK implementation. Please remember you won't be able to retrieve this key once you leave the screen.

Steps to use AAD authentication from SAP using ABAP SDK for Azure

5.1 Creation of RFC destination to Azure Active Directory

Go to transaction SM59 in your SAP system and create new RFC destination of type 'G'. Maintain your Azure Active directory endpoint in the Target host and Event Hub name in path prefix for authorization token as shown below.


Target host: **login.partner.microsoftonline.cn**

Port: 443

Path Prefix: **/<TanentID>/oauth2/v2.0/token**

6 / 11

For Tenant ID details and creating a new tenant id in Azure Active Directory, please refer this document section 'How to setup Azure Active Directory in Azure?'

< 

RFC D

✓

RFC DestinationAZURE_ABAPSDK_AAD

Connection TypeGHTTP Connection to External ServerDescription

Description

Description 1

Description 2

Description 3

Administration

Technical Settings

Logon & Security

Special Options

Target System Settings

Hostlogin.partner.microsoftonline.cnPort443

Path Prefix/ /oauth2/v2.0/token

HTTP Proxy Options

Global Configuration

Proxy Host

Proxy Service

Proxy User

Proxy PW Statusis initial

Now go to 'Logon & Security' tab and choose radio button SSL 'Active' and select SSL certificate 'DFAULT SSL Client (Standard)'.

Do a connection test to make sure it is working. RFC destination is working.

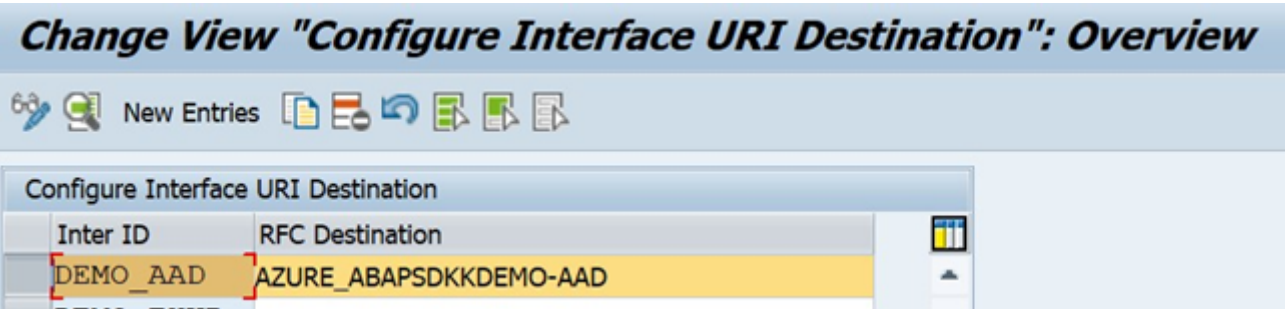
5.2 Configuration

ZREST_CONFIG – Master Table for Interface ID Maintenance. You must define a new Interface name and maintain the RFC destination that was created for target Event hub.

ZADF_CONFIG – This is an Interface extension table. This stores data that is more specific to Azure Services like SAS keys, AAD secrets and processing Method.

ZREST_CONFIG

Create a new Interface ID like 'DEMO_AAD' and Maintain the RFC destination you created earlier.

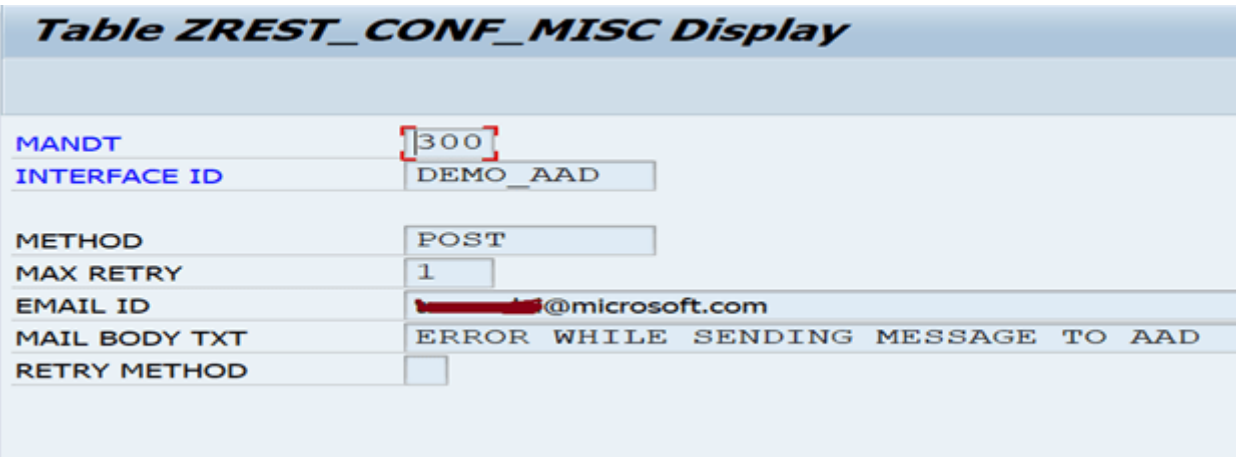


ZREST_CONF_MISC

Create an entry in table 'ZREST_CONF_MISC' for the above interface Id 'DEMO_AAD'.

Details of configuration:

- METHOD is 'POST'.
- MAX_RETRY is number of retry in case of service failure.
- EMAIL_ID is the email id for sending alerts.
- MAIL_BODY_TXT is Text Id to be maintained for the mail content.
- RETRY_METHOD is type of retrial (Regular '0' or exponential '1')



ZADF_CONFIG

Create an entry in table 'ZADF_CONFIG' for the above interface Id 'DEMO_AAD'.

Details of configuration:

- INTERFACE_TYPE is 'Azure Active Directory'.
- SAS_KEY is the shared access key. This is the key which generated in AAD under section 'Generate keys for your application' Step 7 (refer Page 8). You need to change this key in this config table whenever key is changed in Azure.

- URI is left blank. This may be required for future versions.
- SERVICE_TYPE can be synchronous(S) or asynchronous(A)
- IS_TRY is a reprocessing flag, maintain as blank or 'X'. it can be configured for reprocessing in case of failure of services.

Note:This field can be utilized in our future release to control the reprocessing based on value of X. Presently it's should be enabled as blank.

Table ZADF_CONFIG Display

MANDT

300

INTERFACE ID

DEMO_AAD

INTERFACE TYPE

AAD

SAS KEY

URI

SERVICE TYPE

S

IS TRY

ABAP SDK Monitor

We have provided an Interface Monitor (Transaction ZREST_UTIL), using this monitor you can view history of all the messages that were posted to Azure Services. Incase you have a scheduled a background job to post messages to Azure, you can view the statuses of the messages in this monitor. This Monitor can be used for troubleshooting and re-processing of the message as well.

Go to transaction ZREST_UTIL and provide your Interface ID in the selection screen and execute to view all the messages

Monitor

Interface ID

DEMO_AAD

to

Stores Execution Date

08/05/2018

to

08/05/2018

Stores execution time

00:00:00

to

00:00:00

Stores Completed Date

to

Stores Completed time

00:00:00

to

00:00:00

Http Status

to

Message Id

00000000000000000000

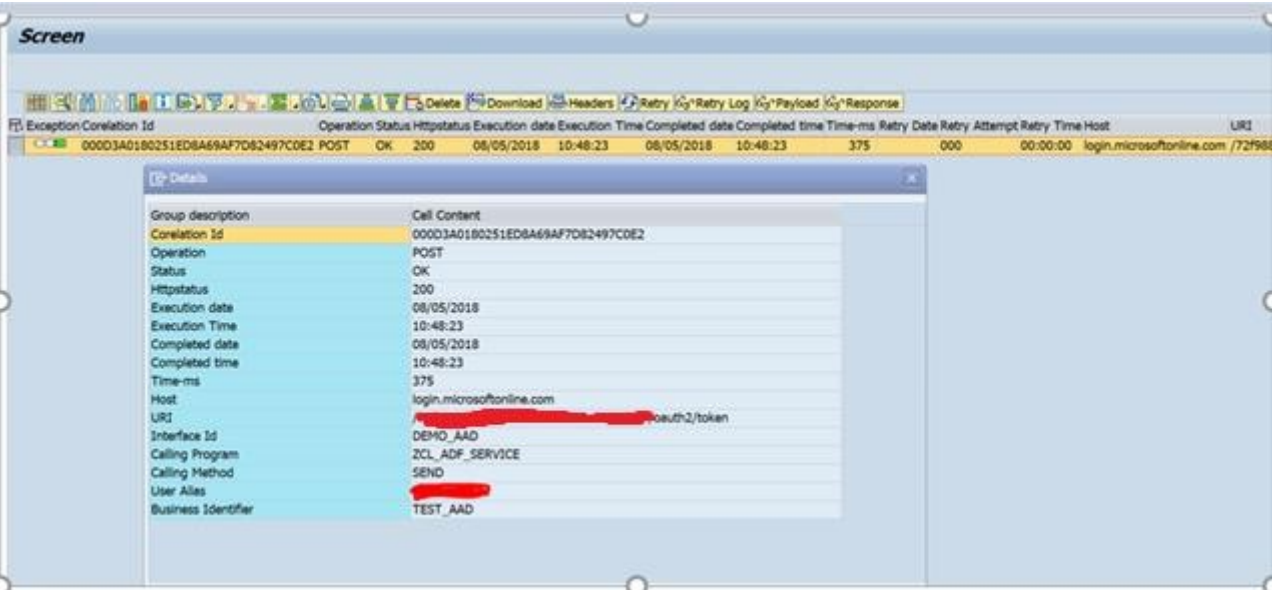
to

00000000000000000000

Business Identifier

to

In this monitor, you can view the status of the HTTPs message and its headers, response, payload and so on. In case of errors, you can also re-process the message from this tool.



Auto re-processing of failed messages

For auto-processing of messages in case of failures, you must schedule a background job for program 'ZREST_SCHEDULER' as a pre-requisite