

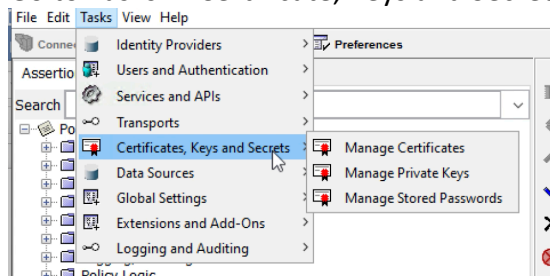
CA SSO and CA APIM Integration Guide Part 2

Version: CA SSO R12.8 SP1, CA API Gateway 9.3, Tool kit 4.2

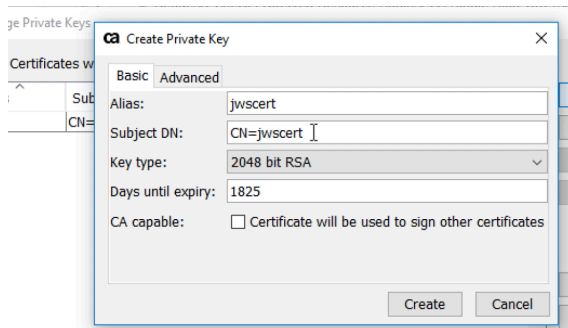
CA SSO configuration – Integration with JWT Authentication

1. Create Private Key for JWT token Signing in APIM Gateway

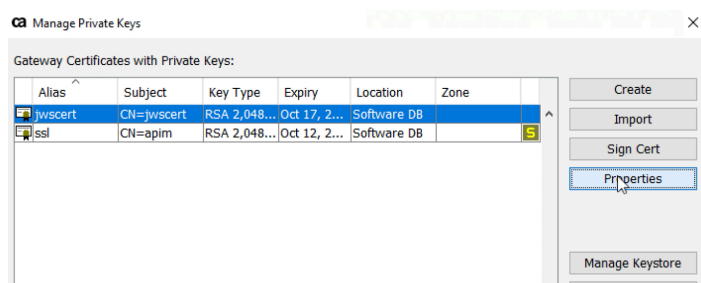
- Login APIM Policy Manager
- Go to Tasks -> Certificate, Keys and Secretes -> Manage Private Keys



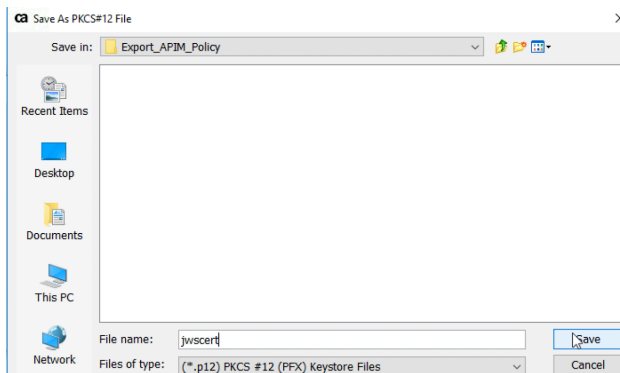
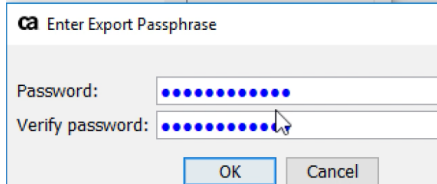
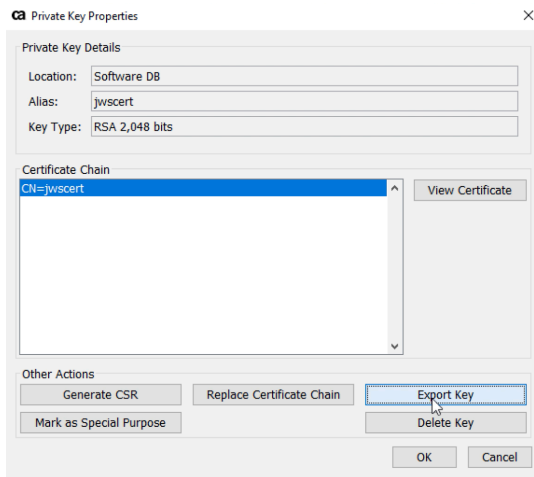
- Create new Private Key. In Alias, input “jwscert”. Then, click “Create” button



- After creating, select newly created key (jwscert) and double click “properties”.

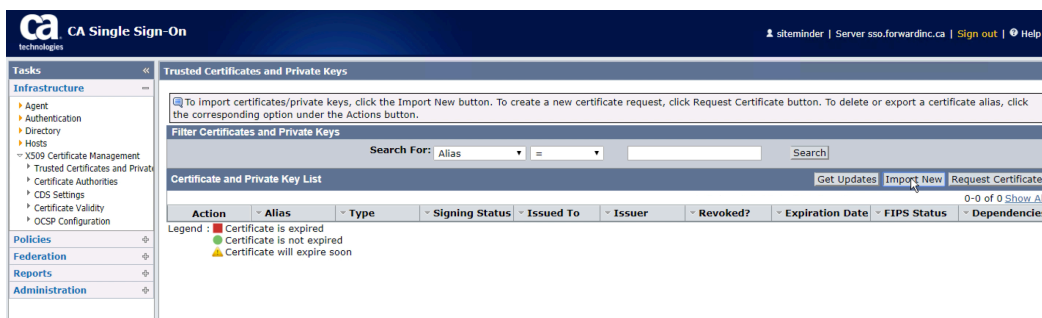


- Click “Export Key” and input the password. (Any password) Then save the exported key file with p12 format.



2. Create JWT Authentication Scheme in CA SSO

- Login CA SSO Admin UI and go to Infrastructure -> X509 Certificate Management -> Trusted Certificates and Private Key. Then, click "Import New"



- Import certificate, which is exported from API Gateway(jwscert). Then, input the password.

1 Select File 2 Password (if needed) 3 Select Entries 4 Confirm

ct a file. PKCS#12 files (private key and certificate) must end in '.p12' or '.pfx'. Any other extension will be trea

Input File As: ☒ Single Key/Cert File ☐ Separate Key File and Cert File
 Use as CA: ☐ Yes ☒ No
 • File: C:\fakepath\jwscert.p12

- After verify the certificate, click “Finish” button.

Trusted Certificates and Private Keys

View Certificates and Private Keys > Import Certificate/Private Key

Back Next Finish Cancel

1 Select File 2 Password (if needed) 3 Select Entries 4 Confirm

Review and confirm selections for import.

Show Details	Alias	Type	Issued To
<input checked="" type="checkbox"/> Hide	jwscert	Private Key and Certificate	CN=jwscert
Issued To: CN=jwscert Issued By: CN=jwscert Valid From: Oct 18, 2018 07:48 AM PDT Key Algorithm: RSA Serial Number: 11:A5:D7:A1:61:31:C7:61 SHA-256 Fingerprint: 82:88:DF:CF:B8:13:17:E2:CF:92:89:EF:2B:2C:B7:47:AB:FE:8B:CB:6C:46:A7:99:C1:70:3F:9C:B6:91:43:42 SHA-1 Fingerprint: E1:7D:C5:55:8F:20:2F:26:9C:0E:6A:17:65:5B:8D:3E:33:35:3D:3B MD5 Fingerprint: 02:71:08:29:6F:3B:6F:69:C1:9C:17:D5:FC:E1:F8:80			

Back Next Finish Cancel

- Go to Infrastructure -> Authentication -> Authentication Scheme -> Create Authentication scheme.
 - Name : JWT Auth Scheme
 - Authentication Scheme Type: JSON Web Token Template
 - User Lookup : sub
 - Certificate Alias List: jwscert

Create Authentication Scheme: JWT Auth Scheme

Authentication Schemes > Create Authentication Scheme: JWT Auth Scheme

• = Required

General

• Name: JWT Auth Scheme Description

Scheme Common Setup

• Authentication Scheme Type: JSON Web Token Template

• Protection Level: 5 [1-1,000,high]

☐ Password Policies enabled for this Authentication Scheme

Scheme Setup

JWT Token Authentication Attribute for Validation

• User Lookup: sub

Certificate Alias List: jwscert

☐ Use JOSE Header KID as Certificate Alias

HMAC Shared Key :

HMAC Confirm Shared Keys:

- Go to APIM Domain (which is created in part 1) and click modify button. Go to Realms and click “Create Realm”.

Modify Domain: APIM

Domains > Modify Domain: APIM

General Realms Policies Responses Rule Groups Variables

Required

Realms

Name	Description	Resource Filter
APIM		/login

Submit Cancel

- Create new Realm
 - Name : JWT Protected
 - Agent: secureproxy-01
 - Resource Filter : /protected/
 - Authentication Scheme : JWT Auth Scheme

Create Realm: JWT Protected

Domains > Modify Domain: APIM > Create Realm: JWT Protected

Required

General

Name: JWT Protected Description:

Domain: APIM

Resource

Agent: secureproxy-01 Lookup Agent/Agent Group

Resource Filter: /protected/

Effective Resource: secureproxy-01/protected/

Default Resource Protection: ☒ Protected ☐ Unprotected

Authentication Scheme: JWT Auth Scheme

- Create Rule
 - Name : GET-POST
 - Resource : *
 - Action : GET, POST, PUT

Required

General

Name: GET-POST Description:

Domain: APIM Realm:

Attributes

Realm and Resource

Resource: *

Effective Resource: secureproxy-01/protected/*

Regular Expression: ☒

Allow/Deny and Enable/Disable

☒ Allow Access ☐ Deny Access

Enabled: ☒

Action

Web Agent actions ☒ Authentication events ☐ Authorization events ☐ Impersonation events

Actions: Get Post Put ProcessSOAP

- Click "OK" button and Submit

Modify Domain: APIM

Domains > Modify Domain: APIM

General Realms Policies Responses Rule Groups Variables

• = Required

Realms Create Realm

Name	Description	Resource Filter
APIM		/login
JWT Protected		/protected/

Submit Cancel

- Go to APIM Domain again and click modify button again. Then, go to policies tab. In policies, click APIM modify.

Modify Domain: APIM

Domains > Modify Domain: APIM

General Realms Policies Resp

• = Required

Policies

Name	Description
APIM	

Create

- In policies, go to Rule tab and click “Add rule” button. Then, select JWT Protected Realm

Modify Policy: APIM

Domains > Modify Domain: APIM > Modify Policy: APIM

General Users Rules Expression

• = Required

Rules

Name

Go Reset

Realm	Rule	Response	Response Group
APIM	APIM login GET Post Rule		

Add Response

Add Rule

Domains > Modify Domain: APIM > Modify Policy: APIM

General Users Rules Expression

Available Rules

• = Required

Rules for APIM

Name

Go Reset

Select	Realm	Rule	Type
<input checked="" type="checkbox"/>	JWT Protected	GET-POST	Rule

Create

- Click OK button and click Submit Button again.

Modify Domain: APIM

Domains > Modify Domain: APIM

General Realms Policies Responses Rule Groups Variables

• = Required

Policies

Name	Description
APIM	

Create

Submit Cancel

3. Create JWT Assertion in APIM Policy Manager

- Login API Gateway Policy Manager and create “Publish Web API”
Service Name : Issue ID Token
Gateway URL: /casso/idthoken*

CA Publish Web API Wizard

Steps

1. Service Information
2. Access Control

Enter the name of the non-SOAP application you want to publish:

Service Name:

Enter the HTTP URL that the Gateway should forward requests to:

Target URL:

Complete the Gateway URL that will receive service requests:

Gateway URL:

Specify the connection and routing information for the non-SOAP application.

Then, click “Finish” button

- Click “Import Policy” and select “Issue ID Token (JWT).xml” file.
- It shows Resolve External Dependencies Wizard. For CA SSO, please select “Change assertions to use this CA Single Sign-On configuration. “CA SSO”. (It is CA SSO configuration, which is created in part 1.)

CA Resolve External Dependencies Wizard

Steps

1. Unresolved CA Single Sign-On Configuration CASSO
2. Unresolved provider CA SSO User DB

Policy contains assertions(s) referring to unknown CA Single Sign-On Configuration

Missing CA Single Sign-On Details

Configuration Name:

Hostname:

Action

☒ Change assertions to use this CA Single Sign-On configuration

☐ Remove assertions that refer to the missing CA Single Sign-On configuration

☐ Import erroneous assertions as-is

Unresolved CA Single Sign-On Configuration CASSO

Then, click next

- In Wizard, select Import erroneous assertion as-is. “Create New Identity Provider”. It will connect CA SSO User Directory.

CA Resolve External Dependencies Wizard

Steps

1. Unresolved CA Single Sign-On Configuration CASSO
2. Unresolved provider CA SSO User DB

Policy contains assertions that refer to an unknown identity provider.

Missing Identity Provider Details

Name: CA SSO User DB

Type: LDAP

Name	Value
userCertificateType	NONE
ldapsearchbase	dc=ForwardInc,dc=ca
ldappwwritebase	

Action

☐ Change assertions to use this identity provider:
☐ Remove assertions that refer to the missing identity provider
☒ Import erroneous assertions as-is

Create new Identity Provider

Unresolved provider CA SSO User DB

Back Next Finish Cancel Help

- It shows “Create LDAP Identity Provider Wizard” and input correct Bind Password. (CA demo123). Using “Test” button, please verify the connection. Depending on the environment, LDAP Host file and other information can be changed.

CA Create LDAP Identity Provider Wizard

Steps

1. Provider Configuration
2. Group Object Classes
3. User Object Classes
4. Advanced Configuration
5. NTLM Configuration
6. Certificate Settings

Provider Type: * GenericLDAP

Provider Name: * CA SSO User DB

LDAP Host URL: * ldap://sso.forwardinc.ca:25389

Move Up

Move Down

Add Edit Remove

☐ Use Client Certificate Authentication: <Default SSL Key>

Search Base: * dc=ForwardInc,dc=ca

Bind DN: uid=superuser,ou=users,ou=northamerica,dc=ForwardInc,dc=ca

Bind Password: •••••••• ☐ Show Password

☒ Allow assignment to administrative roles

☐ Allow updates from:

Write Base: *

Reconnect Timeout: 60000 ☒ Use System Default

This Wizard allows you to configure an LDAP Identity Provider. Fields marked with an asterisk "*" are required.

The "Allow updates from" check box determines whether the LDAP server can be updated starting from the specified write base.

Back Next Test Finish Cancel Help

- Then, click “Next” continually. In User Object classes screen, please verify the Login Name. In CA LDAP, default Login Name should be “uid”. When you are using other user directory, it can be different.

CA Create LDAP Identity Provider Wizard

Steps

1. Provider Configuration
2. Group Object Classes
- 3. User Object Classes**
4. Advanced Configuration
5. NTLM Configuration
6. Certificate Settings

User Object Classes

inetOrgPerson

Object Class Name: inetOrgPerson

Attribute Mapping

Attribute: Mapped to:

User Name: cn

Login Name: uid

Password: userPassword

First Name: givenName

Last Name: sn

Email: mail

Certificate: userCertificate;binary

Kerberos Principal: cn

Kerberos Enterprise Principal:

Add Remove

Then, click Finish button.

- In Resolve External Dependencies Wizard, please select “CA SSO User DB” in Action.

CA Resolve External Dependencies Wizard

Steps

1. Unresolved CA Single Sign-On Configuration CASSO
- 2. Unresolved provider CA SSO User DB**

Policy contains assertions that refer to an unknown identity provider.

Missing Identity Provider Details

Name: CA SSO User DB

Type: LDAP

Name	Value
userCertUseType	NONE
ldapsearchbase	dc=ForwardInc,dc=ca
ldapwritebase	

Action

☒ Change assertions to use this identity provider: CA SSO User DB

☐ Remove assertions that refer to the missing identity provider

☐ Import erroneous assertions as-is

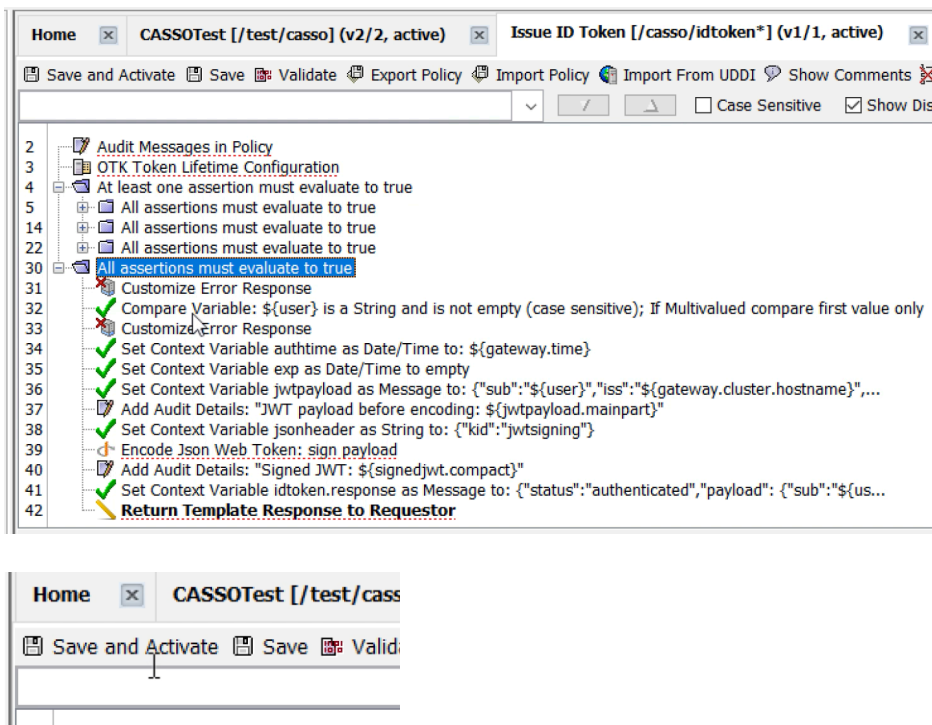
Create new Identity Provider

Unresolved provider CA SSO User DB

Back Next Finish Cancel Help

Then, click “Finish” button.

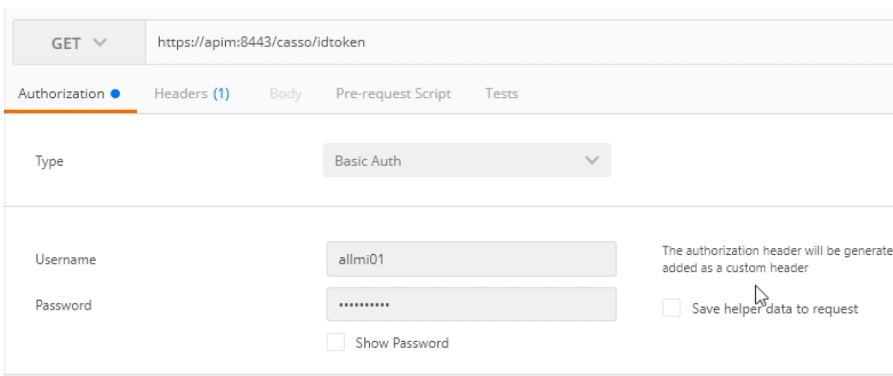
- Then, you can see the imported policies in UI. Click “Save and Activate” icon in UI.



- It shows Resolve External Dependencies Wizard. For CA SSO, please select “Change assertions to use this CA Single Sign-On configuration. “CA SSO”.

4. Test CA SSO and CA API Gateway Integration

- Download PostMan (REST API Client) and execute it.
- In Postman, input the following information.
 - Method: GET
 - URL : **https:// <API GW FQDN>:8443/casso/idthoken**
 - User name: allmi01
 - Password: P@ssword01



Then, click “Send” button.

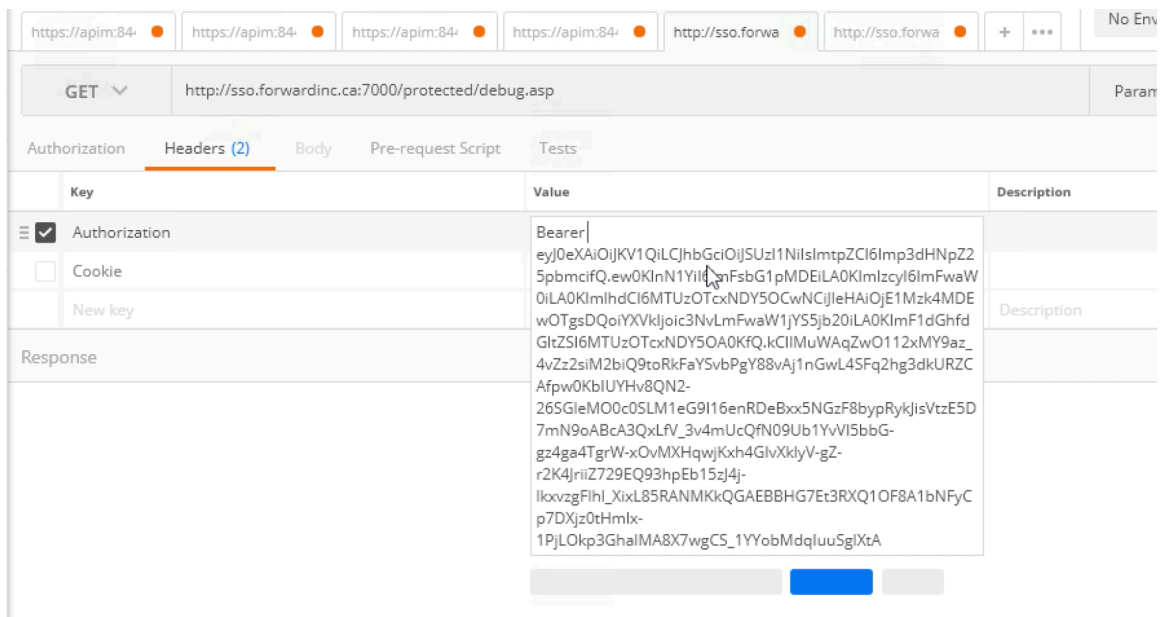
- After successful authentication in API Gateway, it returns JWT token.

```

1 {
2   "status": "authenticated",
3   "payload": {
4     "sub": "allmi01",
5     "iss": "apim",
6     "iat": 1539878468,
7     "exp": 1539964868,
8     "aud": "sso.apimca.com",
9     "auth_time": 1539878468
10  },
11  "jwt": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiIsImtpZCI6ImZ25pbmciFQ...
12 }

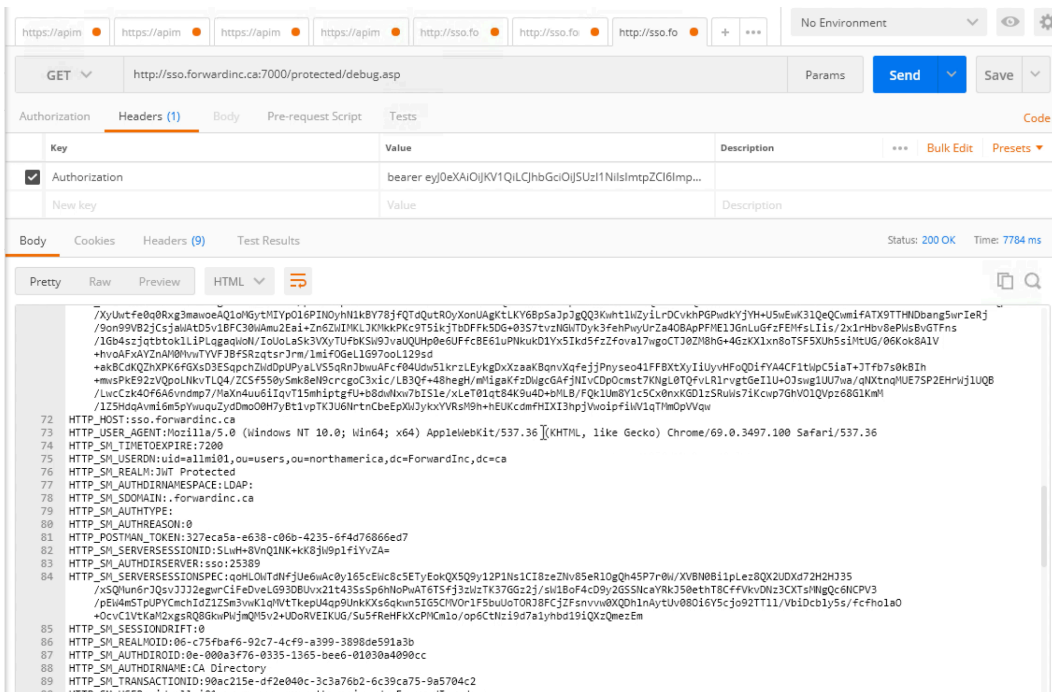
```

- After coping JWT token and sends it to CA SSO protected web page. In Postman client, please input the following information.
 - Method: GET
 - URL : **http://<Access Gateway FQDN>:7000/protected/debug.asp** (Instead of debug.asp, any page can be used as long as it is existed in backed.)
 - Header : Authorization
 - Value : Bearer **<JWT value>** (There is space between Bearer and <JWT value>).



Click "Send" button

- It shows backend result after JWT Authentication.



- After closing Postman client and execute again. It requests to access the resource, which is protected JWT token without providing any token value. It rejects the request.

