FOD Beleid & Ondersteuning
SPF Stratégie & Appui

FAS
-
SAML
Integration guide

Versie 1.1

Table of Contents

1 INTRODUCTION

The objective of this document is to gather the technical information needed to complete the integration with the Federal Authentication Service (FAS).

This document focusses on the support for federal government departments in the implementation and connection of their systems with FOD BOSA IAM. The government departments are Relying Parties who rely on FOD BOSA IAM as Trusted Third Party and as such form a Circle of Trust.

The most important way of connecting to FOD BOSA IAM is the Federal Authentication Service (FAS), My eGov Login. Relying Parties can use this service to identify and authenticate their users. The service can also provide relevant attributes and roles to enable the Relying Party to make an authorization decision.

2 PROCEDURES

2.1 Onboarding Process

2.1.1 Introduction

The onboarding procedure consists of two workflows to follow:

- the first one contains the contractual agreements between FOD BOSA and the customer. The necessary contracts are put in place to describe the trust relationship between the customer and FOD BOSA. Once this is completed and contracts are signed, the next workflow can be started

- the technical onboarding based upon the FOD BOSA onboarding document. During this step, the technical implementation will occur to make the FAS environment ready for the customer. The customer will, together with FOD BOSA, prepare the onboarding specification.

2.1.2 Specification

The specifications of the Relying Party will be documented in an onboarding document. This document describes the technical details of the trust between FAS and Relying Party and contains the metadata of the Relying Party, the desired attributes, role information, etc. This trust is based upon the SAML-protocol standard where FOD BOSA is the Identity Provider (IdP) and the customer the Service Provider (SP). A basic knowledge of this protocol is required by the customer to have a good understanding of the requested information in the document.

This document is the base of trust between FAS and the Relying Party.

2.1.3 Configuration

The configuration of the FAS is executed after completing the onboarding document. The FAS and all necessary components are configured to support the trust relation between FAS and the Relying Party.

The configuration is first executed in the FOD BOSA Integration environment (INT). After successful testing the configuration can be migrated to the production environment (PROD). The customer can send their development, integration and acceptance setup to be tested within the FOD BOSA integration environment.

2.2 Certificate Request Process

When using SAML it is common to sign SAML messages. This implicates for a Relying Party the creation of a signing key X.509 with accompanying certificate.

This signing key can be obtained in two ways:

- Or you can buy a certificate at Certificate Authorities, which implies a certain cost.
- Or you can create this certificate yourself within Unix, Linux Apache or via online SAML tools.

Here are some links where you can get more information: Unix, Windows:

http://www.ibm.com/support/knowledgecenter/SSFKSJ_7.5.0/com.ibm.mq.sec.doc/q012770_.htm
Linux Apache : https://www.linux.com/learn/creating-self-signed-ssl-certificates-apache-linux SAML Tools : https://www.samltool.com/self_signed_certs.php

Please note that this certificate is only used for security reasons

3 CUSTOMER (ORGANIZATION)

3.1 Name

The name of the customer (or organization).

3.2 KBO#

The number of the company as registered in the Belgian company-register, namely KBO.

3.3 Name integrator (when applicable)

The name of the integrator in case your implementation is executed by a third-party integrator.

4 CONTACT INFO

For the contact-info, you must make a distinction between the project-based start-up and the operational modus afterwards.

Start-up is the onboarding itself in project modus, where web application, together with link to FAS are developed by the customer itself or via a third-party system integrator.

After the successful completion of the onboarding, the onboarding gets into operational modus. The operational contacts are the people who need to be aware of FAS maintenance, scheduled downtime, updates and releases etc.

4.1 Start-up

4.1.1 SPOC

Please mention the name of the person who will be the key-contact at the customer or the integrator, responsible to answer all possible questions, related to the onboarding. We advise to have someone who is aware of internet-technologies, web-based applications, development, SAML and role-based federation.

This person will be called the SPOC. He/she can delegate the technical questions to the project manager, functional Analyst or architect.

The SPOC is not always experienced to pick-up all the roles mentioned above. Therefore he/she can assign some operational persons, to perform one of these roles:

Project manager Functional Analyst Architect

These persons need to be defined, together with their contact details.

4.1.2 Project manager

The project manager is dedicated by the SPOC to take the lead of the onboarding-project at customers' side

4.1.3 Functional Analyst

The functional analyst will define the business requirements and align them with the technical implementation of the onboarding.

4.1.4 Architect

This is the technical person responsible for the onboardings environment like platform, metadata, security-level, firewall, authentication-methods, metadata etc.

4.2 Operational

4.2.1 Service Owner

The person who is holding the onboarding as business-service (ITIL)

4.2.2 Service Manager

The person who is responsible that the FAS service is available to the customer.

4.2.3 Security Manager

The person who is responsible for the internet-protocol based implementation like protocols, firewall, VPN, authentication strategies etc.

4.2.4 Service Desk

Contact details of the service desk of the customer.

5 APPLICATION

5.1 Name

Name of the application that the customer will use.

In case an integrator uses his platform to house the application of the customer, then it's advisable to mention both e.g., IntegratorX – ApplicationY where "IntegratorX" is the name of the platform or integrator, and "ApplicationY" the name of the application.

5.2 Purpose

A small description of the application, can assist FOD BOSA in defining if the requirements mentioned by the customer in the onboarding document, are correct. If not, FOD BOSA will deliver some guidance to optimize the onboarding.

5.3 Audience

This describes by which audience, the application will mainly be used: this can be citizen, enterprise, government (which is also enterprise)

- Citizen: the application will be used for public services and mainly be used by citizens using authentication means like eID, Token, Authenticator App etc.
- Enterprise: the application will be used mainly by enterprises internally, or inter-enterprise
    - Internally: the enterprise-application will use the FAS-integration to define different roles to access the application. These roles will make their application role-and level-based. A company can decide how roles can be requested:
        - Direct
        - Role-Invitation
    - Inter-enterprise: this means that the application is public but will be used by enterprises only. These enterprises can be grouped together, based upon their company-number. An example is Tax-on-Web which is used by accountant-offices.
- The application can of course use both citizen and enterprise if required

5.4 Roles required (Y/N)

An application can be onboarded in two ways:

Free of roles: the application will only use FAS attributes and service, without any defined roles or role-attributes (N)

Role-based: the application will use roles and optionally attributes (J)

5.5 URL's application

Please mention all the URLs used by the application, defined in the different environments like Test, Development, Integration, Acceptance and Production.

5.6 Used Component (SAML)

Please mention the used component to provide the SAML information. Some of the possibilities are:

- OpenAM,
- Drupal,
- PHP Module,
- Java Lib,
- ADFS
- …others.

6 AUTHENTICATION MEANS

To access the FOD BOSA IAM Services, there are several ways of authentication possible: Via:

- eID
- Recognised electronic identifier (partner): itsme® and myID.be®
- Commercial Certificate
- Token
- Authenticator App (TOTP)
- SMS (OTP)
- Username + Password

Depending on the target audience of the application (citizen / enterprise). these authentication methods are translated in SAML into “context” and “contract”. A contract is a technical SAML definition to define the requested authentication method.

FAS works with “Levels of Assurance”. Individual contracts per authentication mean are still supported but will be phased out.

This means all authentication means equal or higher than the level sent by the authentication request will be available. Setup is done by adapting the SAML authentication request parameters.

RequestedAuthnContext to Comparison="minimum" and

AuthnContextClassRef to urn:be:fedict:iam:fas:‘Target group’:Levelx00

Example

....
<saml2p:RequestedAuthnContext Comparison="minimum">

<saml2:AuthnContextClassRef
xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">urn:be:fedict:iam:fas:enterprise:Level300</saml2: AuthnContextClassRef>
....

Below you find a table, containing the SAML overview of all by FOD BOSA offered and supported authentication means:

Security level	Authentication Means	Authentication contract in SAML2
High	eID **(1)**	urn:be:fedict:iam:fas:'Target groups':Level500
	itsme® **(1)**	urn:be:fedict:iam:fas:'Target groups':Level450
	myID.be® Authenticator App Mail OTP SMS OTP **(2)**	urn:be:fedict:iam:fas:'Target groups':Level400
Low	Token **(3)**	urn:be:fedict:iam:fas:'Target groups':Level300
	Username/Password	urn:be:fedict:iam:fas:'Target groups':Level200
Without identification	Self-registration without usage of NRN	urn:be:fedict:iam:fas:'Target groups':Level100

(1) The digital keys, **eID and itsme®, must be present** at each onboarding.
If the customer chooses a certain level, the keys with a higher technical level must also be offered as well.
(e.g.: the customer chooses level 400, then the FAS screen displays the keys of levels 400, 450 and 500).

(2) SMS OTP: the optional digital key.
The customer pays the cost of sending the SMS when authenticating with this key. This requires an agreement between the customer and the mobile operator, which charges the cost directly to the customer.
If you are interested in the offer of this digital key 'code by SMS', please ask BOSA for the corresponding annex.

(3) The token is in 'phase out' mode.
No new tokens can be activated by end users since 25/09/2020.

Where "Target groups" supported are:

Target groups	Values
	citizen
	enterprise

It is important to request the correct authentication contract as this is used to provide attributes and roles. If the necessary contract isn't included in the request, this method isn't shown during authentication.

A user is always authenticated in a certain target group with an authentication mean. The possible target groups are citizen and enterprise.

An example of contract(s) in a SAML authentication request looks as follows:

```
<saml2p:AuthnRequest
AssertionConsumerServiceURL="https://iamapps.belgium.be/cma/fedletapplication"
                Destination="https://idp.iamfas.belgium.be/fas/SSORedirect/metaAlias/idp"
                ForceAuthn="false"
                ID="a3ch9g8j69ii3i9i4g1741ag3fja36j"
                IsPassive="false"
                IssueInstant="2017-05-04T12:50:00.255Z"
                ProtocolBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
                Version="2.0"
                xmlns:saml2p="urn:oasis:names:tc:SAML:2.0:protocol"
                >
   <saml2:Issuer
xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">https://iamapps.belgium.be/cma</saml2:Issuer
>
   <saml2p:RequestedAuthnContext    Comparison="minimum">
      <saml2:AuthnContextClassRef
xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">urn:be:fedict:iam:fas:citizen:Level300</saml2:A
uthnContextClassRef>
   </saml2p:RequestedAuthnContext>
</saml2p:AuthnRequest>
```

7 ATTRIBUTES

7.1 Attribute Service

The FOD BOSA IAM Service also contains a service that can provide some additional attributes, embedded in the SAML response, which can be used to decide upon the authorization to the application. This service is called the Attribute Service (AS).

The exchanged attributes are defined on FAS for each Service Provider (during the onboarding). There is a distinction between context aware attributes and context-free attributes.

This service can deliver attributes related to private information of the principal or related to the role the principal achieved.

7.2 Standard Personal Attributes

7.2.1 NameID

The NameID code is a unique identifier for the principal during a session. FAS expects a transient NameID. This value changes each session per principal.

7.2.2 FedID

A unique value for the principal within a certain context. This value can be used as a unique identifier without the use of the national registry number.

7.2.3 Target Group (Context)

Three values: citizen, enterprise or all.

7.2.4 National Registry Number

The national registry number of the principal, namely RRN.

7.2.5 Given Name

The given name of the principal.

7.2.6 Last Name

The surname of the principal.

7.2.7 *Preferred Language (SMA)*

This is the language which the principal defined in his SMA User Interface.

7.2.8 *Preferred local language*

The preferred language of the principal as registered in FOD BOSA IAM.

7.2.9 *Personal email address*

The personal email address as registered in FOD BOSA IAM.

7.2.10 *Authentication Method*

Describes the authentication method used. This allows applications to offer different functionalities based on authentication method.

7.3 Role Attributes

These attributes hold all the role information for the principal specific for the Relying Party. The information is base64 encoded and has an XML structure.

The XML format uses a scheme to deliver the role information with a set attribute “role-name” and one or more possible parameters.

```
<rol:RoleResult     xmlns:rol="http://be.fedict.rolemgmt/RoleXMLSchema">
        <rol:Role name="<application_role>" xmlns:role="http://be.fedict.rolemgmt/RoleXMLSchema">
        <rol:RoleAttribute name="CompanyId">999999999</rol:RoleAttribute>
        <rol:RoleAttributename="FEDictDomain">SOMEDOMAIN</rol:RoleAttribute>
        </rol:Role>
</rol:RoleResult>
```

The above information is available as a base64 encoded value in the SAML response in the roles attribute.

The above role information results in the following value:

PHJvbDpSb2xlUmVzdWx0IHhtbG5zOnJvbD0iaHR0cDovL2JlLmZlZGljdC5yb2xlbWdtdC9Sb2xlWE1MU2NoZW1hIj4NCgk8cm9sOlJvbGUgbmFtZT0iPGFwcGxpY2F0aWVfcm9sPiIgeG1sbnM6cm9sPSJodHRwOi8vYmUuZmVkaWN0LnJvbGVtZ210L1JvbGVYTUxTY2hlbWEiPgkJCQk8cm9sOlJvbGVBdHRyaWJ1dGUgbmFtZT0iQ29tcGFueUlkIj45OTk5OTk5OTk8L3JvbDpSb2xlQXR0cmlidXRlPg0KCQk8cm9sOlJvbGVBdHRyaWJ1dGUgbmFtZT0iRkVEaWN0RG9tYWluIj5TT01FRE9NQUlOPC9yb2w6Um9sZUF0dHJpYnV0ZT4NCgk8L3JvbDpSb2xlPg0KPC9yb2w6Um9sZVJlc3VsdD4=

7.3.1 *Standard additional Role attributes*

During role definition, the customer can request 3 additional attributes which are related to the role, namely OrganizationID, RefOrganizationID and professional email-address.

7.3.1.1 *OrganizationID*

The company number of the company will be returned.

7.3.1.2 *RefOrganizationID*

The company number of the master company A. will be returned e.g., the number of Company A which delegates services to their accountant company B.

7.3.1.3 *Professional email-address*

The professional email address of the logged-in user will be returned, e.g., the accountant's company's email address

7.3.2 *Non-standard additional Role Attributes*

During role-onboarding, the customer can define and request additional attributes that can be added to the specific role.

8 METADATA

8.1 Metadata URL's of Identity Provider (FOD BOSA)

The standard URLs to add in the metadata are (based upon environment):

Omgeving	URL
Integration	https://iamapps-public.int.belgium.be/saml/fas-metadata.xml
Production	https://iamapps-public.belgium.be/saml/fas-metadata.xml

In these jsp files, the authentication contracts, mentioned in chapter 6 can/must be added.

8.2 Metadata of Service Provider (Client)

For each URL/application that the client wants to get access to FAS, metadata is expected in xml-format. Here are some guidelines to follow when creating your metadata for SAML 2.0:

8.2.1 Example:

```
<EntityDescriptor
        xmlns="urn:oasis:names:tc:SAML:2.0:metadata" ID="I5bf4e6f87a221b0a"
        xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
        xmlns:ds="http://www.w3.org/2000/09/xmldsig#" entityID="<entityID of RP >"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="urn:oasis:names:tc:SAML:2.0:metadata http://docs.oasis-
        open.org/security/saml/v2.0/saml-schema-metadata-2.0.xsd">

<SPSSODescriptor AuthnRequestsSigned="true"
protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
        <KeyDescriptor>
                <ds:KeyInfo>
                        <ds:X509Data>
                                <ds:X509Certificate>
                                        <Certificate of RP used for signing SAML messages (must be
                                issued by a CA with a public OCSP)>
                                </ds:X509Certificate>
                        </ds:X509Data>
```

```
                </ds:KeyInfo>
        </KeyDescriptor>
        <SingleLogoutService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
        Location="<Endpoint Location for Single Logout Requests>" ResponseLocation="<Endpoint
        Location for Single Logout Responses (Optional)>" >
        </SingleLogoutService>
        <NameIDFormat>urn:oasis:names:tc:SAML:2.0:nameid-format:transient</NameIDFormat>
        <AssertionConsumerService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
        Location="<EndPoint Location for HTTP POST SAML Assertion Messages" index="0"
        isDefault="true"> </AssertionConsumerService>
        <AssertionConsumerService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"
        Location="<EndPoint Location for HTTP Redirect SAML Assertion Messages" index="1"
        isDefault="true">
        </AssertionConsumerService>
</SPSSODescriptor>
</EntityDescriptor>
```

8.2.2 *Explanation*

8.2.2.1 *Global*

Make sure that for each SAML-attribute a closing item is added e.g., <EntityDescriptor has as closing counterpart </EntityDescriptor>. The closing counterpart starts always with </

8.2.2.2 *EntityID*

The entityID must be unique for every Relying Party and is specified in the entitydescriptor-paragraph.

It can be the url of the application or any unique identifier that contains preferably the name of the organization and application.

Example's:

- fediam.minfin.fgov.be (applicationname.organisation)
- btb.csam.be (applicationname.organisation)
- https://kbo-bce-wi.economie.fgov.be/kbo/ (url of the application)

8.2.2.3 SPSSODescriptor

The Service Provider SSO Descriptor contains:

- the AuthenticationRequestSigned attribute, which preferably must be set to true,
- the certificate
- the Single Logout Service
    - Binding where there are the following options available in SAML 2.0:
        - SAML SOAP Binding (Limited support, only for artifact)
        - http Redirect Binding
        - http Post Binding
        - http Artifact Binding
    - Location: Endpoint Location for Single Logout Requests
    - ResponseLocation: Endpoint Location for Single Logout Responses (Optional)
- NameIDFormat: only transient is supported
- AssertionConsumerService Binding:
    - this contains the type of binding
    - and the location

9 REQUEST AND RESPONSE EXAMPLES

9.1 Authentication

In the simplest use case, the Service Provider asks a certain AuthenticationContext for a session of the principal and the FAS delivers certain identity attributes, no privilege information to the Service Provider.

A typical request looks as follows. The important attributes are highlighted. **ForceAuthn**, **Issuer**, NameIDPolicy and **RequestedAuthnContext**.

```
<saml2p:AuthnRequest
AssertionConsumerServiceURL="https://iamapps.belgium.be/cma/fedletapplication"

Destination="https://idp.iamfas.belgium.be/fas/SSORedirect/metaAlias/idp
                    " ForceAuthn="false"
                    ID="a3ch9g8j69ii3i9i4g1741ag3fja36j"
                    IsPassive="false"
                    IssueInstant="2017-05-04T12:50:00.255Z"
                    ProtocolBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
                    POST" Version="2.0"
                    xmlns:saml2p="urn:oasis:names:tc:SAML:2.0:protocol"
                    >
    <saml2:Issuer
xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">https://iamapps.belgium.be/cma</sa
ml2:Issuer>
    <saml2p:RequestedAuthnContext Comparison="minimum">
        <saml2:AuthnContextClassRef
xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">urn:be:fedict:iam:fas:enterprise:L
evel300</saml2:AuthnContextClassRef>
    </saml2p:RequestedAuthnContext>
</saml2p:AuthnRequest>
```

A typical response would look as follows. The important attributes are highlighted.

- Combination of Destination, ID and InResponseTo are used to prevent replay attacks and abuse of captured responses.
- The Status Code
- The AuthContext
- The content of the attribute statement (all information within <saml:AttributeStatement>)

```
<samlp:Response xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
Consent="urn:oasis:names:tc:SAML:2.0:consent:obtained"
Destination="https://sp2.iamdemo.be:443/fedlet/fedletapplication"
ID="idEgYuquDWiDQypa6K1U0JaSk-x8s"
InResponseTo="s27092295f77b7224177ef3c1c3ad7038ec635ab9a" IssueInstant="2011-10-
12T19:00:15Z" Version="2.0">
```

9.2 Logout

A typical LogoutRequest looks as follows:

```
<samlp:LogoutRequest xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
ID="s2d1cdf8aa8b3063edb026b7ce86efda77ed52417b"
Version="2.0"
IssueInstant="2015-01-28T19:35:27Z"
Destination="https://idp.iamfas.belgium.be/fas/IDPSloRedirect/metaAlias/idp"
>
<saml:Issuer
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">https://iamapps.belgium.be/</saml:I
ssuer>
<saml:NameID
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
NameQualifier="https://idp.iamfas.belgium.be/fas"
SPNameQualifier="https://iamapps.belgium.be/"
Format="urn:oasis:names:tc:SAML:2.0:nameid-format:transient"
>myNs4TJ12S7I6OW3gGk6uA7Fvb/a</saml:NameID>
<samlp:SessionIndex
xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol">s21d268460527f0c526f9591e3398642d04
85ba112</samlp:SessionIndex>
</samlp:LogoutRequest>
```

A typical LogoutResponse looks as follows:

```
<samlp:LogoutResponse xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
ID="s63732ba858c3fd3c394faf58f3fa4833de1a9727"
Version="2.0"
IssueInstant="2015-01-28T19:35:27Z"
Destination="https://iamapps.belgium.be/fedletSloPOST"
InResponseTo="s2d1cdf8aa8b3063edb026b7ce86efda77ed52417b" >
<saml:Issuer
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">https://idp.iamfas.belgium.be/fas</
saml:Issuer>
<samlp:Status xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol">
<samlp:StatusCode xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
Value="urn:oasis:names:tc:SAML:2.0:status:Success"
/>
</samlp:Status>
</samlp:LogoutResponse>
```