Finesse SSO (Single Sign-On) Design Document

Revision History

Date of creation/ revision	20-12-2016
Date of Next Revision	DD-MM-YY

Version No.	Published Date	Sections Changed	Authored By	Reviewed By
1.0	[redacted]	Created	Javed Shaikh	
2.0	[redacted]	Modified		
3.0	[redacted]	Modified		
4.0	[redacted]	Modified	Javed Shaikh	
4.1	[redacted]	Modified	Javed Shaikh	

Distribution

This document has been distributed to:

Name	Name

Table of Contents

1

Background

Finesse application uses SAML as the standard technology for SSO Integration. The SAML based components have the ability to work with IDPs that have different login mechanism and therefore different credentials, databases and backend implementations.

1.1 Frontend Perspective

1.1.1 Existing flow

When a user has to login into Finesse then

A. Click on Finesse application URL
B. Enter internal username/password of Finesse
C. On click of submit the Finesse application home page appears

1.1.2 Proposed flow (SP-initated Approach)

Finesse will provide the SSO integration using the SP- initiated SSO method. When a SSO user has to login into Finesse then

A. Click on Finesse application URL
B. If user is already logged in into IDP, the home page of Finesse appears
C. If user is not logged in into IDP then the Single Sign-on page(login page of IDP) appears and on submission of page home page of Finesse appears

1.1.3 Exceptions

When a Non-SSO user has to login into Finesse, there will be a separate URL. Only Non-SSO users will be able to access application using this URL.

1.2 Backend Perspective

Finesse application has its own user management mechanism. In order to enable sign in into the application via IDP users, mapping of users is required.

A. The mapping of all applicable IDP users' emails will be done with Finesse users. It means a user in Finesse will have email which will belong to IDP.
   [NOTE]: The default unique ID which will be mapped with Finesse User is email. In case a different unique ID is received, this will need to be managed using a custom ID handler.

B. When SSO user clicks on Finesse URL, if he is already logged in into IDP, the authentication of logged in user will be done and on successful authentication page will be redirected to Finesse home page.
C. When the user clicks on Finesse URL, If he is not logged in into IDP, the login screen will appear where user has to put his SSO credentials and on successful authentication, the Finesse home page will appear.
D. Any failure of Authentication at IDP level, the login page of IDP will appear.
E. For Non-SSO users, existing login screen of Finesse will be applicable
F. Any SSO user can't access Finesse via Finesse login page.

2

Resources, Standards, APIs

2.1 SAML: Security Assertion Markup Language

An XML-based open standard for exchanging authentication and authorization between an identity provider (a producer of assertions) and a service provider (a consumer of assertions).

2.1.1 Service Provider

The business application which is supposed to use credentials of IDP for login. In this case Finesse Application is the Service Provider. It has its own user management mechanism.

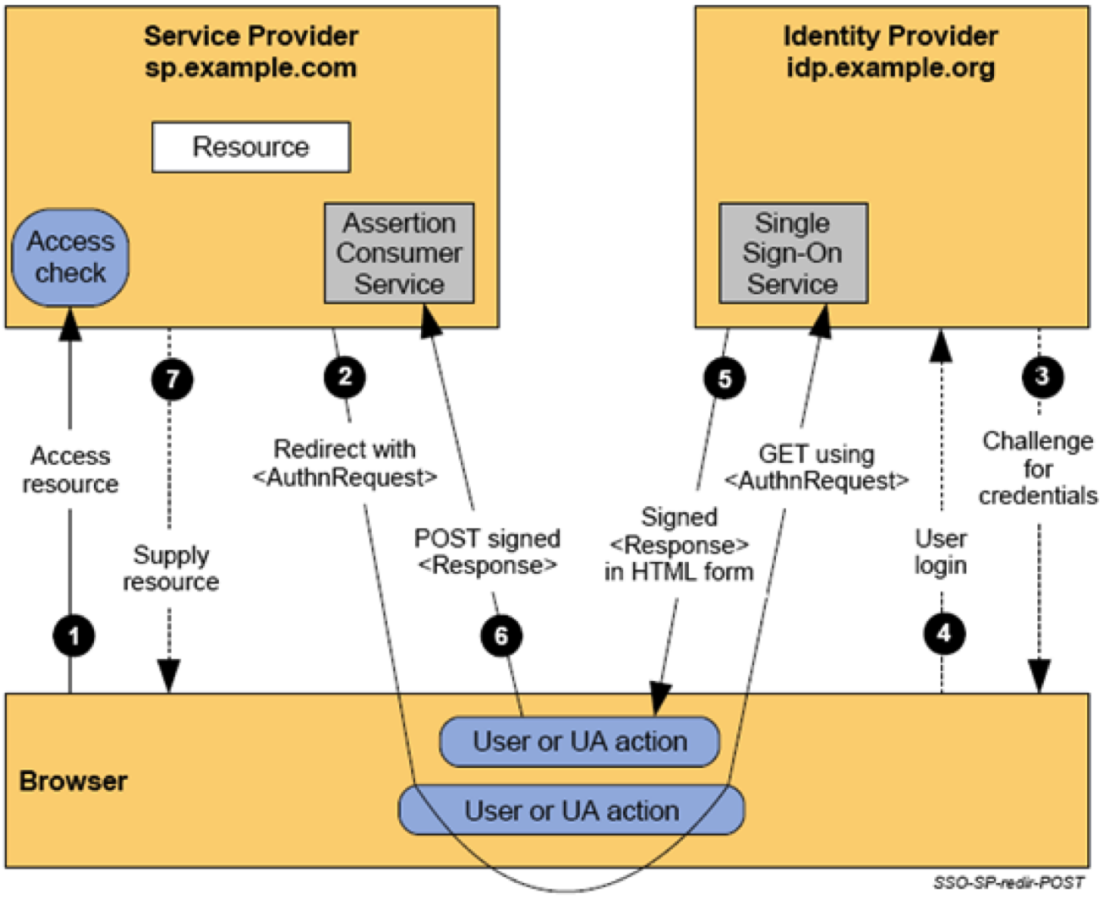
2.1.2 Identity Provider

It contains SSO login page and username/password store.

Note: The Finesse application supports SAML based IDP. The application is tested with Microsoft Azure based IDP.

2.1.3 How it works?

When a user visits any Service Provider (SP), it is redirected to the Identity Provider (IDP) in order to login. Then, the Identity Provider asserts to the SP that you are authenticated. The IDP may also tell the Service Provider about any interesting attributes of the authenticated user.



2.2 Technical Implementation

2.2.1 Identity Provider

The metadata of IDP will be required to integrate into Service Provider application. It is an xml file which has information about end point url, keys etc.

Note: The IDP metadata file should be accessible from Finesse application

2.2.2 Service Provider

- Integrate Spring security jars
- Integrate Security SAML jars
- Create Service Provider metadata file.
- Integrate Spring Security XML file and change the configuration according to requirement
- Create/Modify supporting Java classes/JSP pages to handle authentication

3

SAML SSO Authentication workflow

3.1 IDP Users

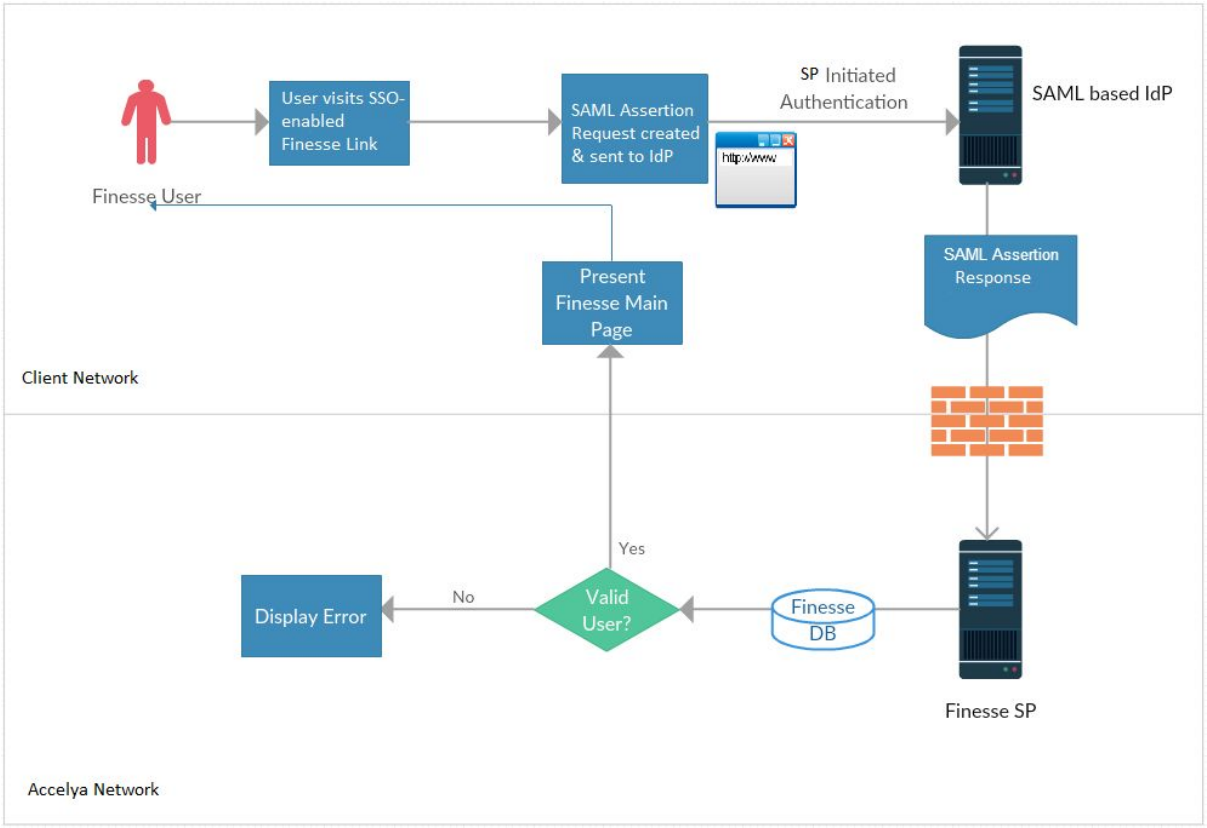
Below are the steps for login into Finesse by users who are registered into IDP.

A. The user will access SSO-enabled URL configured in the application (Finesse) for using the Spring SAML Extension.
B. The application will generate a new authentication request using SAML 2.0 protocol, digitally sign it and send it to the IDP. After authentication at IDP with the user's account, he/she will be redirected back to the application.
C. The [redacted] will be done at Finesse database
D. On successful authentication at IDP, Finesse application will check mapped Finesse user for given email(Unique ID) into Finesse database and the user will be logged in with mapped Finesse user.

3.2 Finesse Users

Below are the steps for login into Finesse by users who are not registered into IDP. In this case [redacted] will belong to these types of users

A. The user will access different URL then SSO-enabled URL which will redirect to Finesse application login page(existing Finesse Login screen)
B. The [redacted] mechanism and will be redirect to application home page on successful login


User visits SSO-enabled Finesse Link
SAML Assertion Request created & sent to IdP
SP Initiated Authentication
SAML based IdP
Finesse User
Present Finesse Main Page
SAML Assertion Response
Client Network
Yes
No
Valid User?
Display Error
Finesse DB
Finesse SP
Accelya Network

4

Detailed Authentication workflow

4.1 Login via IDP (SSO Login)

4.1.1 Successful authentication at IDP

On click of Finesse SSO-enabled URL,

A. If user is already logged in into IDP, the user will be authenticated with IDP and redirected to Finesse application home page
B. If user is not logged in into IDP, the Single Sign on screen(login screen of IDP) appears and on entering the username/password page will be redirected to Finesse home page
C. If Finesse application session is timed out and user is logged in into IDP the user will be redirected to the landing page. This landing page is an html page with valid information/error message with SSO URL Link.

4.1.2 Failed authentication at IDP

On click of Finesse SSO-enabled URL,

A. The Single Sign on screen appears. If user enters invalid username/password, error message is displayed on Single Sign on screen.
B. On submit of credentials in Single Sign on screen, if the given user is not mapped with any Finesse user then the user will be redirected to landing page with valid error/information message.
C. On submit of credentials in Single Sign on screen, if the mapped Finesse user is not active then the user will be redirected to landing page with valid error/information message.
D. On submit of credentials in Single Sign on screen, if user is active at SSO but password is expired in Finesse then the user should not be allowed to login. The user should will be redirected to landing page with valid error/information message.

4.2 Login via Finesse (for users)

The login via Finesse will work as per existing behavior except below

A. The SSO users can’t login via Finesse

A flag will be [redacted] table which will denote if user is SSO user or not.

B. In the cases where SSO is down, the ADMIN user will be allowed to login via Finesse(Non-SSO)

5

Global and local logout

5.1 Global logout

If user has logged out from IDP, he will be logged out from Finesse application too. The user will be redirected to landing page with valid error/information message.

5.2 Local logout

The behavior of Local logout will be same as Global logout. On local logout, the user will be redirected to landing page with valid error/information message.

6

Update Users in Finesse

The Finesse application has feature to update data from GUI using MDCF. This feature will be used to load the SSO user data into Finesse. The format of MDCF file will be finalized and shared with Client. The Administrator user can upload user data using the MDCF file.

7

Open Questions

1. Is the SSO solution SAML 2.0 compliant?
2. Will AA prefer the SP-initiated SSO or the iDP initiated SSO for Finesse?
3. What will be the test environment to use?
4. Who is the SSO vendor for the Client IdP?
5. What will be the security parameters that we need to adhere to?

8

Assumptions

1. Finesse will provide the SSO integration using the SP- initiated SSO method
2. HTTP Post binding will be the underlying mechanism for SAML assertions