



# HOW TO DELEGATE USER AUTHENTICATION

---

**Distribution:** Internal/External  
**Author(s):** R&D / PLM Support  
**Recipients(s):**  
**Cc:**  
**Date created:** March 23, 2015  
**Date of last update:** March 23, 2015  
**Version:**  
**Subject:** [Commentaires ]

---

---

## Contents

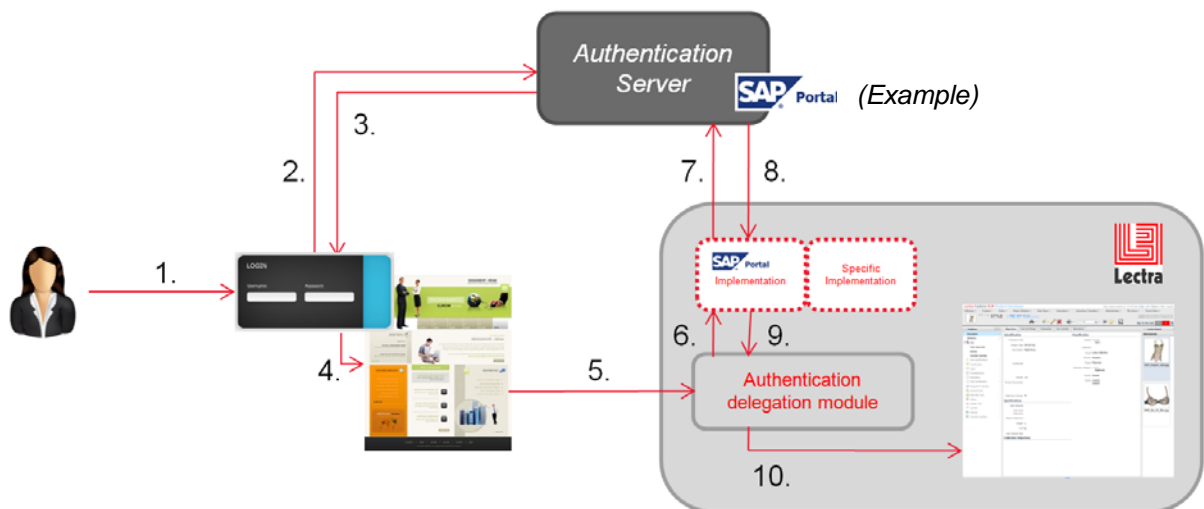
1. Overview .....	3
2. Custom delegate implementation.....	4
2.1 Lectra Fashion PLM SDK.....	4
2.2 Build & Deploy.....	5
2.3 Checking user delegation.....	5
2.3.1 Checking the delegate.....	5
2.3.2 Accessing to the Lectra Fashion PLM through the delegate.....	6

## 1. OVERVIEW

Single sign-on is a process that allows network users to access all authorized network resources without having to log in separately to each resource. Single sign-on allows you to validate user authorization against your corporate user database or other client applications rather than having to separate user passwords managed by Lectra Fashion PLM.

Lectra Fashion PLM authentication delegation is based on a web module in charge of communication with an authentication server. As all authentication servers have their own approach/technology to handle user authentication, the module is customizable to fit your IT requirements.

It works as described below:



1. A user logs into the main application (e.g. portal) using the current login and password
2. User and password are checked by the authentication server
3. If the User is authenticated, a token is sent back to the login module
4. Access to the application is granted
5. From the application, any hyperlink to the Lectra Fashion PLM will go through the authentication module: the user, token and Lectra Fashion PLM URL (e.g. <http://<my.server>/pdm>) will be sent to this module
6. This information will be sent to the implementation corresponding to your Authentication server
7. The authentication server is contacted to check the user and token validity
8. It sends its response
9. The authentication delegation module receives the response
10. If the user and token are validated, a session is created on the Lectra fashion PLM and the application is opened without prompting the user for its credential.



Users must be declared and enabled in the authentication server and in the Lectra Fashion PLM.

## 2. CUSTOM DELEGATE IMPLEMENTATION

### 2.1 Lectra Fashion PLM SDK

To implement your specific authentication delegation, use the Lectra Fashion SDK.

In the package `com.lectra.sdk.fip.authentication.delegation`, you have to create a class implementing the interface `com.lectra.platform.authentication.delegation.domain.models.AuthDelegate` which is composed of only 2 methods:

- String `getId()` : return the name of the delegate implemented (e.g.: `SAPPortalDelegate`)
- `AuthStatusBean check(String user, String token)` : return an `AuthStatusBean` (a boolean indicating that access is granted or not, and a message)

A basic example of implementation is provided in the Lectra Fashion PLM SDK.

```
package com.lectra.sdk.fip.authentication.delegation;

import com.lectra.platform.authentication.delegation.application.AuthDelegationException;
import com.lectra.platform.authentication.delegation.domain.models.AuthDelegate;
import com.lectra.platform.authentication.delegation.domain.models.AuthStatusBean;

/**
 * Sample AuthenticationDelegation
 * Only users whose name starts with "Fred" or "Nazha" and with token
 * equals to the length of the user name are authorized
 * AuthenticationDelegationExample class
 * @author LECTRA
 */
public class AuthenticationDelegationExample implements AuthDelegate {

    @Override
    public AuthStatusBean check(String user, String token)
        throws AuthDelegationException {
        if(user != null && (user.startsWith("Fred") || user.startsWith("Nazha"))) {
            if(token != null && token.equals(String.valueOf(user.length()))) {
                return new AuthStatusBean(true, "");
            }
        }
        return new AuthStatusBean(false, "User not authorized with this token");
    }

    @Override
    public String getId() {
        return "AuthenticationDelegationExample";
    }
}
```

In this example, only requests for authentication that match these two conditions will be successful

- Name must start with "Fred" or "Nazha"
- Token is a numeric equal to the name length (number of characters in the name). For instance . "Frederic" with a token equal to 8

The user must be defined and enabled in the Lectra Fashion PLM.

Every implementations of authentication delegation must be declared in the file

```
conf/com.lectra.platform.authentication.delegation.domain.models.AuthDelegate
```

This file will contain the name of the Classes corresponding to the implementations.

(e.g.: com.lectra.sdk.fip.authentication.delegation.AuthenticationDelegationExample)

## 2.2 Build & Deploy

To build the custom delegate, run the ant task "build" provided in the Lectra Fashion PLM SDK.

It will generate the file `auth-delegation-custom.jar` in the folder:

```
<PLM SDK install folder>\ws_sdk\<Your Customization Project>\build\archive.
```

This jar must be copied **on each Lectra Fashion PLM application server** in the same folder as the jar used for the Custom Business Rules:

```
<PLM Fashion Install>\externalized-customization\lib
```

Note: this task is done automatically by the customization installer that you can build with the ant task "build-installer" (or "release" in V4R1).

See PLM SDK User guide for more information on this installer.

The application server must be restarted after the installation of the delegate.

## 2.3 Checking user delegation

### 2.3.1 Checking the delegate

You can use a REST client (search for "Rest client" on the internet) to check that the delegate is properly implemented.

Call the delegate with the following URL: (using the GET method)

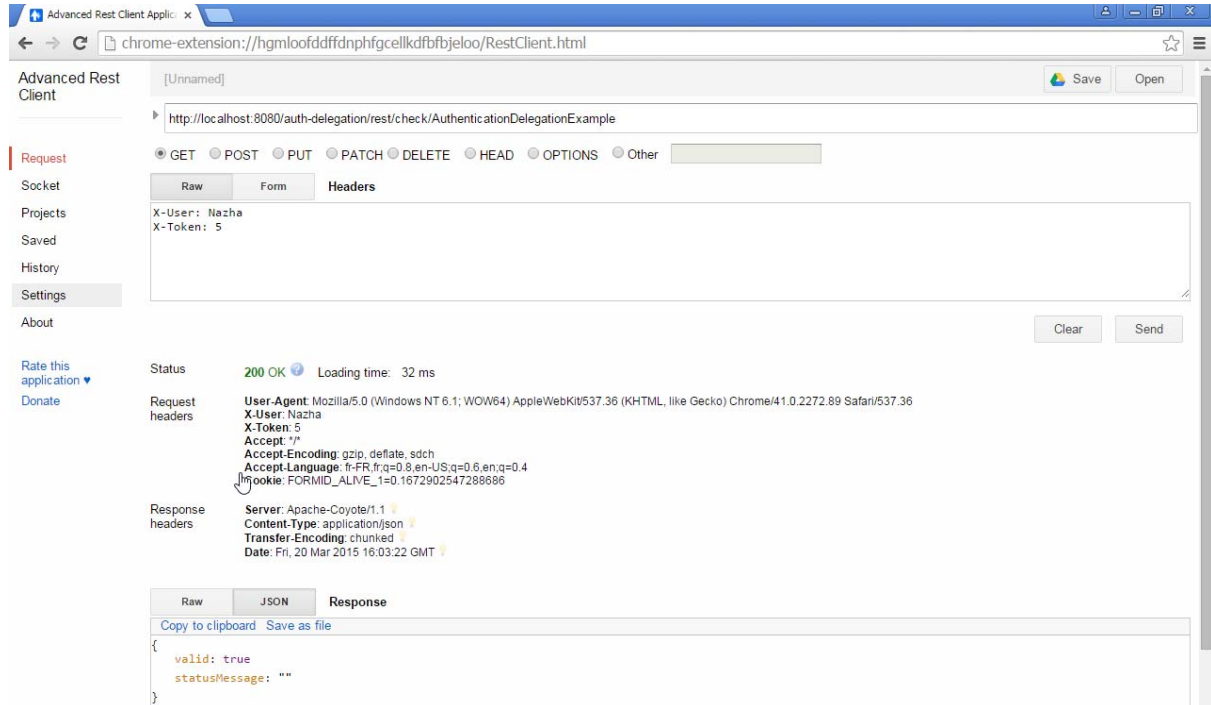
```
http://<plmserver>:8080/auth-delegation/rest/check/<yourDelegate>
```

(e.g.: `http://localhost:8080/auth-delegation/rest/check/AuthenticationDelegationExample`)

and set in the header of the HTTP request the following properties:

X-User : Name of a user

X-Token : a String corresponding to a token authenticating the user



### 2.3.2 Accessing the Lectra Fashion PLM through the delegate

To access the Lectra Fashion PLM through the SSO, you have to call the following REST resource (method GET):

`http://<plmserver>/authentication/rest/sso/<yourDelegate>`

and set the following properties in the header of the http request:

X-User: Name of a user

X-Token: a String corresponding to a token authenticating the user

X-PLM-URL: the URL of the PLM you want to access

The SSO resource will

- Check if the user X-User is known and enabled in PLM
- Call the authentication delegate system `delegateId` to check if the token X-Token is valid for the user X-User
- If everything is ok, redirect the request to X-PLM-URL with a valid `FIPSession` parameter, which provides a direct access to the requested PLM page without PLM login

### Example

Advanced Rest Client

[Unnamed]

▶ <http://wnbouchia/authentication/rest/ssl/AuthenticationDelegationExample>

GET
  POST
  PUT
  PATCH
  DELETE
  HEAD
  OPTIONS
  Other

Headers

X-User: Fred  
 X-Token: 4  
 X-PLM-URL: <http://wnbouchia/pdm/Home.displayHome.wbx?>

Rate this application ▼  
 Donate

Redirect #1 To: <http://wnbouchia/pdm/Home.displayHome.wbx?&FIPSession=-2189491989722415121> with status: 303 HTTP/1.1 303 See Other

Redirection information has not been cached.

Date: Fri, 20 Mar 2015 16:41:11 GMT  
 Server: Apache/2.2.11 (Win32) mod\_ssl/2.2.11 OpenSSL/0.9.8i mod\_auth\_sspi/1.0.4 mod\_jk/1.2.28  
 Access-Control-Allow-Origin: \*  
 Access-Control-Allow-Headers: origin,accept,content-type  
 Access-Control-Allow-Credentials: true  
 Access-Control-Allow-Methods: GET, POST, PUT, DELETE, OPTIONS, HEAD  
 Access-Control-Expose-Headers: Location  
 x-responded-by: cors-response-filter  
 Location: <http://wnbouchia/pdm/Home.displayHome.wbx?&FIPSession=-2189491989722415121>  
 Content-Length: 0  
 Keep-Alive: timeout=5, max=100  
 Connection: Keep-Alive  
 Content-Type: text/plain

Redirect #2 To: <http://wnbouchia/pdm/Home.displayHome.wbx?FORMID=9&FIPSession=-2189491989722415121> with status: 302 HTTP/1.1 302 Moved Temporarily

Redirection information has not been cached.

Date: Fri, 20 Mar 2015 16:41:11 GMT  
 Server: Apache/2.2.11 (Win32) mod\_ssl/2.2.11 OpenSSL/0.9.8i mod\_auth\_sspi/1.0.4 mod\_jk/1.2.28  
 Expires: Mon, 26 Jul 1997 05:00:00 GMT  
 Last-Modified: Fri, 20 Mar 2015 05:41:11 GMT  
 Cache-Control: no-cache, must-revalidate  
 Pragma: no-cache  
 CALLERFORMID:  
 FORMID: 9  
 Location: <http://wnbouchia/pdm/Home.displayHome.wbx?FORMID=9&FIPSession=-2189491989722415121>  
 Vary: Accept-Encoding  
 Content-Encoding: gzip  
 Content-Length: 20  
 Keep-Alive: timeout=5, max=99  
 Connection: Keep-Alive  
 Content-Type: text/html; charset=UTF-8; charset=UTF-8

Status **200 OK** Loading time: 97 ms

Request headers  
 X-User: Fred  
 X-Token: 4  
 X-PLM-URL: <http://wnbouchia/pdm/Home.displayHome.wbx?>  
 Host: Access-Control-Allow-Headers: origin,accept,content-type  
 Access-Control-Allow-Credentials: true  
 Access-Control-Allow-Methods: GET, POST, PUT, DELETE, OPTIONS, HEAD  
 Access-Control-Expose-Headers: Location  
 x-responded-by: cors-response-filter  
 Location: <http://wnbouchia/pdm/Home.displayHome.wbx?&FIPSession=-2189491989722415121>  
 Content-Length: 0  
 Keep-Alive: timeout=5, max=100  
 Connection: Keep-Alive  
 Content-Type: text/plain