Valid - Integration

Quick Start Guide for developers

Valid

The unexpected Strong Authentication

March 2015

1. Index

2. Introduction

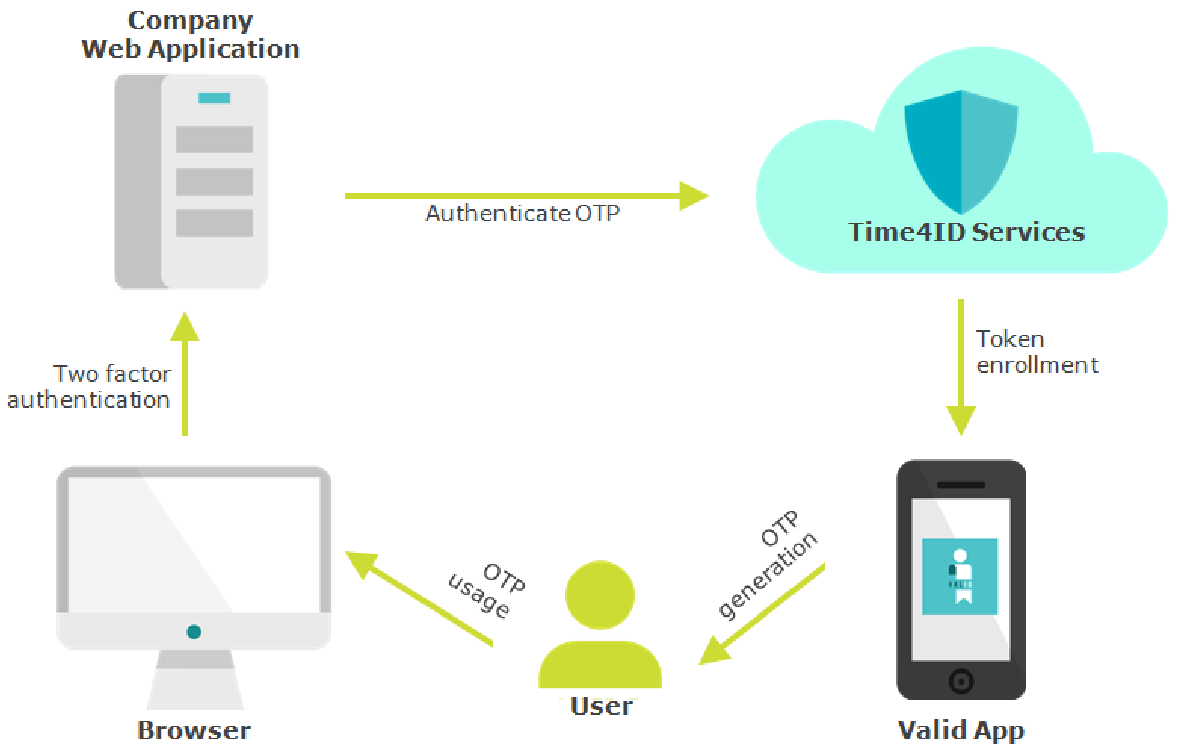
Valid is a solution designed for Small and Medium Enterprises, based on Time4ID, the cloud platform for two factor authentication.

Time4ID is a strong authentication solution that enables the use of OTP (One Time Password) in cloud mode minimizing the cost of deployment. The platform enables users to simultaneously run different OTP providers in a transparent way. In addition to the native software OTP (Time4ID) it supports hardware OTP from leading vendors (Vasco, RSA, ...) and OTP via SMS for mobile legacy.

The software OTP are available as an App for mobile devices (iOS and Android) and as SDK for application integration. In addition to traditional explicit OTP, Time4ID offers implicit OTP, OTP signature and out of band OTP in push mode. The innovative encryption of the seed, its protection on HSM (Hardware Security Module) and the possibility to add data fraud, make the solution very safe.

3. Overview

As described above with Time4ID is possible to realize many solutions, meeting a lot of different requirements. Valid is one of these possible implementations, especially designed for a fast and easy integration.



Architecture components are:

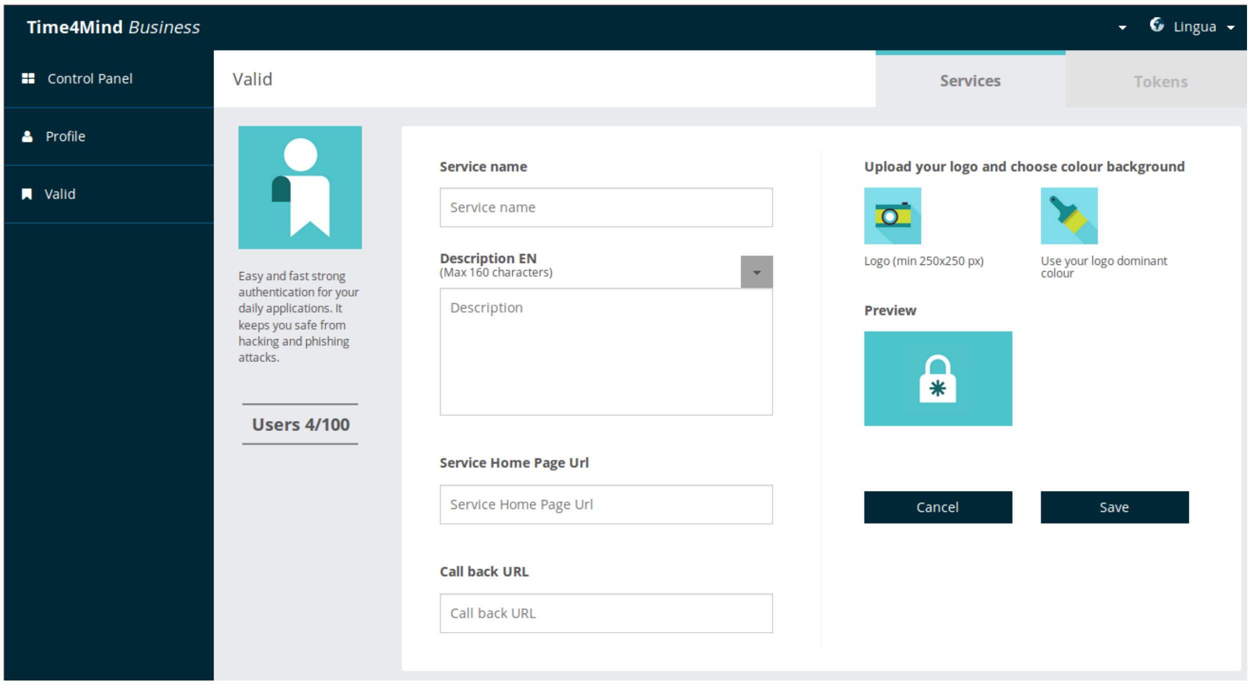
- Web Application: your service that you want to secure using Valid;
- Browser: the desktop or mobile interface for the user to access your service;
- Valid App: the mobile App that your user will install on the smartphone;
- Valid “Widget”: the web pages that drive the user along the enrollment process;
- Time4ID: the two factor authentication backend service that:
  - enroll tokens on the App of your users;
  - verify the OTP generated by users on their App and used to access on your web applications

Before to go on it's better to become confident with the process and have a clear vision of every step. For this reason i's very important to test the user experience at least once, using the demo site available here: http://demo.time4mind.com

4. Service setup

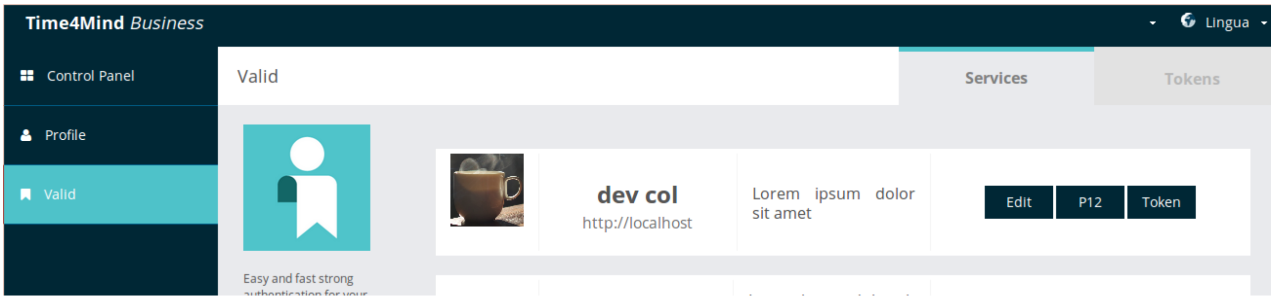
The admin control panel (https://admin.time4mind.com) is a tool designed for developers and sysops that need to configure their services. During the registration on the control panel you create a "Company Account" to which you can associate as many services you need.

In the Valid section of the control panel you have the "Add Service" button to create a new configuration dedicated to your web application.



The most important field, from a technical point of view, is the "Callback URL". This URL must point to the webpage of your web application, where the user will arrive at the end of the enroll procedure.

The last step to complete the configuration is the generation of the SSL credentials (private key and certificate), necessary to use the API in the most secure way.



When you press the P12 button on the service card, the control panel will request your login password again, for security, to grant you access to the P12 management area.

The control panel will generate a P12 file containing your SSL credentials, protected with a new P12 dedicated password that you have to choose and remember (not to be confused with your *login* password!). Don't forget it or you will have to generate a new P12. This because we do not store your private key: the P12 you download is the unique copy we can provide you, for security reason. If you lose the file or the password, you have to generate a new P12.

P12 is the extension of an archive file format for storing cryptographic objects, defined by the PKCS#12 standard (http://en.wikipedia.org/wiki/PKCS_12). PFX is the extension used by Microsoft, but the file format is now the same, so you can simply rename the extension.

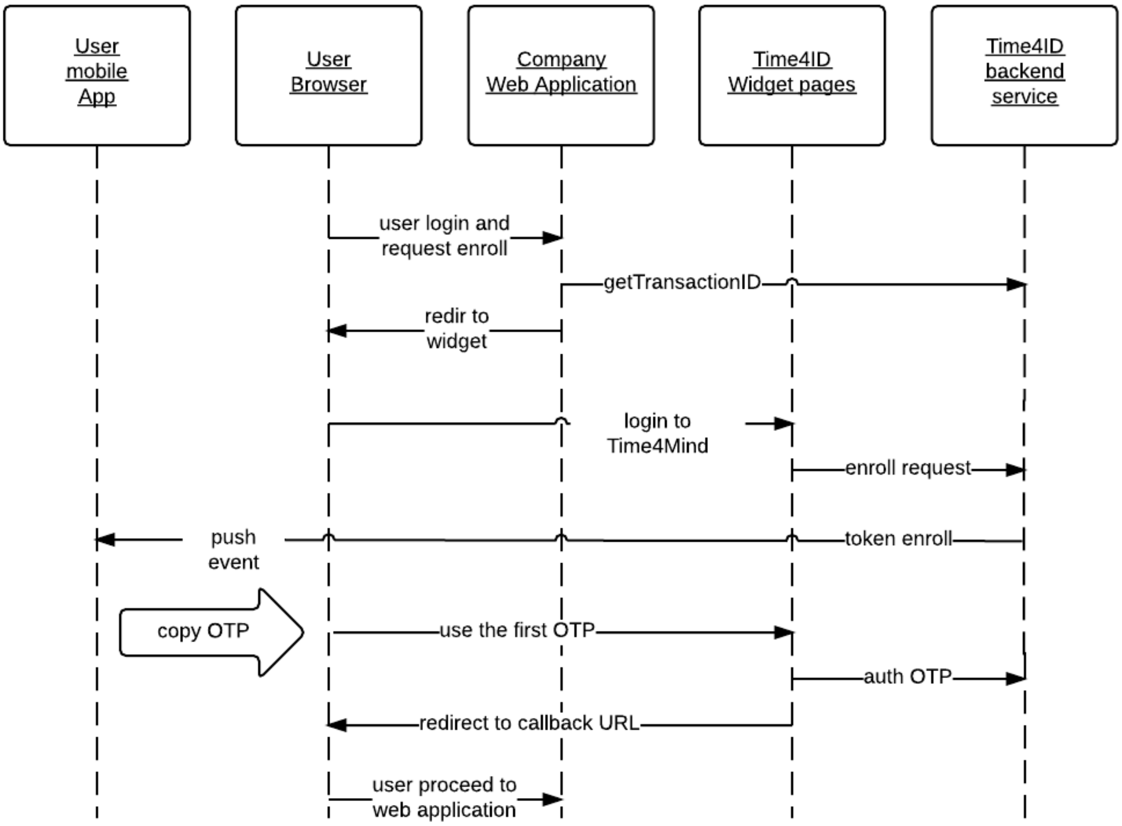
Depending on the development language and library used, in some cases it's better to have the PEM file format instead P12 or PFX. PEM file format use the base64 encoding for the binary objects (key and certificate). The example provided at GitHub (https://github.com/Valid-2FA) contain a script to make the conversion from P12 to PEM file format.

5. Token enrollment

The enrollment is the process of delivering to the user the token (the cryptographic quantity from which the one-time-passwords are derived).

The steps are described below:

1. the user login to your company web application;
2. the user asks to activate the two factor authentication for his account;
3. the company web application initiates the process requesting a TransactionID to Time4ID backend service; then redirect the user to the "Widget" URL;
4. the user is driven from the "Widget" to download the mobile App, register herself and enroll the token; finally the "Widget" requires the user to insert the first OTP generated by the Valid App, to be sure that the process is completed correctly;
5. the "Widget" redirects the user to the "Callback URL" you configured for your service.



The method getTransactionID generates a uniq TransactionId associated with a temporary access rule. This rule defines the access as belonging to the company account who made the request. You must specify the company username (extUserId). This method is available via JSON-RPC 2.0 at the URL:

https://sme.time4mind.com/Time4UserServices/services/backend/t4ujson

Input:

Name	**Type**	**Description**
extUserId	string	uniq identifier of user in the original Company context (i.e. email or username)

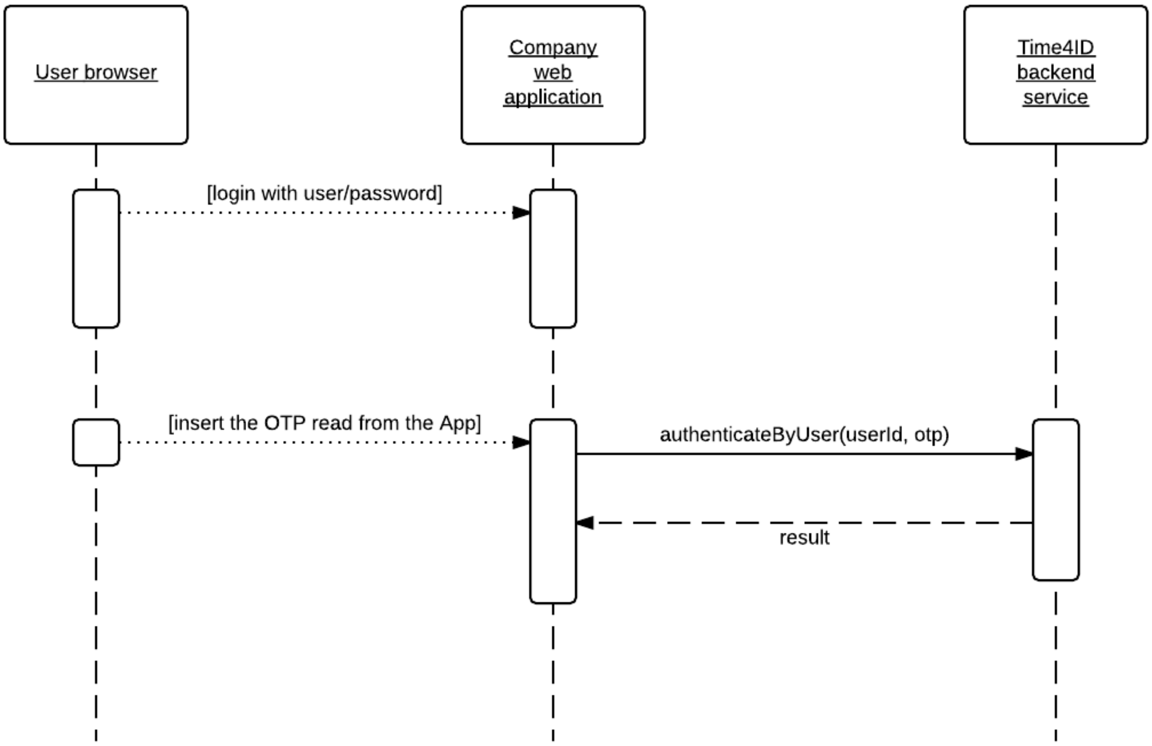
Output:

Name	**Type**	**Description**
transaction Id	string	temporary Id to associate the user to the Company

The Valid Widget usage is very easy. You have to redirect the browser with a GET method to the URL https://valid.test4mind.com/serv/getCompanyServiceInfo.php?tid=<transactionId>&extUID=<extUserId>
where values <transactionId> and <extUserID> are respectively the output and the input of the getTransactionID API previously used.

6. OTP authentication

Now your web application can ask the user to insert an OTP to authorize some accesses. The authentication is performed with just a simple call to Time4ID backend.



The method authenticateByUser verifies an OTP associated to a given company username (extUserId). This method is available via JSON-RPC 2.0 at the URL:

https://sme.time4mind.com/Time4eIDv2/backend/auth

Input:

Name	Type	Description
extUserId	string	uniq identifier of user in the original Company context (i.e. email or username)
otp	string	one time password to authenticate

7. Error codes

Here are listed the most common error codes useful for a proper management.

In case of an error the result is an object Error with two parameters:

Name	Type	Description
code	integer	error code
message	string	error message

getTransactionID

Code	Message
257	param externaUserId can not be empty
768	error when try to load data from database
17154	user in not attested to any node
17155	node authorization failed
17664	service non found on nodeId where user is attested

authenticateByUser

Code	Message
808	the user entered a bad OTP
816	user entered 3 bad OTP in a row, use the App to resync
809	token is locked, it must be unlock using the admin control panel https://admin.time4mind.com
-4002	no token found for this user, her should go to the enroll process

Widget

Value	Message
257	The service is not available or authorized (TransactionID is empty)
401	An unexpected error occurred on the server, retry later
402	Service data are not available, retry later
403	The company user has already activated the service with a different Time4Mind credentials

404	The Time4Mind user has already activated the service with a different company username
405	The service has reached the maximum number of users allowed
406	SESSION EXPIRED OR NOT VALID
515	Time at your disposal is expired! (TransactionID expired)
516	TRANSACTION ID IS NOT VALID
1001	Well done! The service is now enabled

8. History

Date	Author	Description
13 March 2015	EC	First release
17 March 2015	EC	Completed error codes tables