

SELF SERVICE RESET PASSWORD MANAGEMENT
WEB INTERFACE GUIDE

*Copyright © 1998 - 2020 Tools4ever B.V.
All rights reserved.*

No part of the contents of this user guide may be reproduced or transmitted in any form or by any means without the written permission of Tools4ever.

DISCLAIMER - Tools4ever will not be held responsible for the outcome or consequences resulting from your actions or usage of the informational material contained in this user guide. Responsibility for the use of any and all information contained in this user guide is strictly and solely the responsibility of that of the user.

*All trademarks used are properties of their respective owners.
www.tools4ever.com*

Contents

1. Introduction 1

2. Installing the Web Interface 1

- 2.1. Installing the SSRPM COM Object.....1
- 2.2. Installing the Web Interface2
 - 2.2.1. Requirements2
 - 2.2.2. MSI Installation3
 - 2.2.3. Manual Installation3
 - 2.2.4. Configuration4
- 2.3. Configuring SSL.....5
- 2.4. Configuring the Web Interface in a DMZ6
- 2.5. Troubleshooting7

3. Web.config 7

- 3.1. Log8
- 3.2. SSRPM Website9
- 3.3. CIV Website27

4. Web Interface Overview 34

- 4.1. Main Window34
- 4.2. Enrollment34
- 4.3. Forgot my password35
- 4.4. Change password35
- 4.5. Forgot my username35
- 4.6. Manage your information in Active Directory35
- 4.7. Onboarding36
 - 4.7.1. Import37
 - 4.7.2. Language39
 - 4.7.3. Wizard39

5. Frequently Asked Questions (FAQ) 40

6. Index 41

1. Introduction

This document describes the SSRPM web interface. The chapter Installing the Web Interface covers the installation of the SSRPM Web Interface with Microsoft IIS. The chapter Overview covers the pages that are available with the SSRPM Web Interface.

Please note that the Web Interface shipped with SSRPM can be fully customized. Refer to the "COM Object Guide" for a complete description of the SSRPM COM Object used in the Web Interface.

2. Installing the Web Interface

This chapter describes step by step how to install the web interface. This document assumes that IIS is already installed and running. The Web interface is installed in three steps:

1. Installing the SSRPM COM Object. First, the SSRPM COM object must be installed on the machine running IIS.
2. Installing the Web Interface. Second, the Web Interface must be copied to a directory on the machine running IIS.
3. Configuring IIS. Last, IIS must be configured so that the web interface can be accessed.

2.1. Installing the SSRPM COM Object

The Web Interface uses the SSRPM COM Object to access the SSRPM Service. This COM Object must first be registered on the machine running IIS. This chapter will describe in detail how to register the SSRPM COM Object. Please note that the ASP.NET web interface requires the multi-threaded COM object.

1. Copy the SSRPM COM Object to the target machine. The SSRPM COM Object is installed together with the SSRPM Admin Console and can be found in the directory "C:\Program Files\Tools4ever\SSRPM\COM". There are 2 versions of the COM object: 32-bit and 64-bit. Use the 32 bit version for 32-bit operating systems and the 64-bit version for 64-bit operating systems. The name of the files are "SSRPMCOMMT.dll" and "SSRPMCOMMTx64.dll", respectively.
2. Also copy the file 'RegisterMT.bat' to the same directory as the SSRPM COM Object.
3. Go to the machine running IIS.
4. Double click on the file 'RegisterMT.bat'. This command need to be executed with Administrative rights. If the registration succeeds, the following dialog is displayed:

This completes the installation of the SSRPM COM Object. The next chapter will describe how to install the Web Interface on the machine running IIS.

2.2. Installing the Web Interface

The SSRPM Web Interface is a fully functioning interface that users can use to enroll into SSRPM, reset passwords or unlock accounts. This chapter describes how to install the SSRPM Web Interface on the machine running IIS.

If the web interface is also intended to be used by the client (enrollment wizard) you will need to setup 2 separate web interfaces. One with Forms Authentication and one with Windows Authentication. This will be explained later on in this chapter.

You only need the web interface with Windows authentication if you are going to use the enrollment wizard and you want the user to be automatically logged on.

1. The SSRPM Web Interface files are installed together with the SSRPM Admin Console. They can be found in the directory 'C:\Program Files\Tools4ever\SSRPM\Admin Console\Examples\Client Web Interface'. Select all of the files in that directory (including any subdirectories) and copy them to an empty directory on the machine running IIS.

The installer also includes the CIV Web Interface which is a interface that the help desk can use to verify the identify of callers.

The installation is very similar to the SSRPM web interface, the only notable difference is that web interface requires 'Windows Authentication'.

1. The CIV Web Interface files are installed together with the SSRPM Admin Console. They can be found in the directory 'C:\Program Files\Tools4ever\SSRPM\Admin Console\Examples\CIV Web Interface'. Select all of the files in that directory (including any subdirectories) and copy them to an empty directory on the machine running IIS.

There are 2 methods to install the web interface(s).

1. Manually, see chapter: *Manual Installation*.
2. Using the MSI package, see chapter *MSI Installation*.

Please note that the web interface by default is configured to connect to the SSRPM Service on the same machine as IIS. This behaviour can be changed by changing the value of the setting "server" in the web.config file.

2.2.1. Requirements

The system requirements of SSRPM web interface are:

Operating Systems:

- Windows Server 2008 R2
- Windows Server 2012

Software requirements:

- .NET 4.5

2.2.2. MSI Installation

The file 'SSRPMWebInterfaceSetup.msi' can be used to install the web interface(s) and can be found in the directory 'C:\Program Files\Tools4ever\SSRPM\Admin Console\'.
By default the installation procedure will install 2 web interface (one with forms authentication and 1 with Windows authentication).

Optionally, you can also install the Helpdesk Caller Identity Verification (CIV) web interface.

It will create new web interface entries in IIS, *SSRPM_FA* for forms authentication, *SSRPM_WA* for Windows authentication and/or *SSRPM_CIV* for the helpdesk CIV web interface. By default they will listen to the ports 80, 81, 800, respectively.

During the installation procedure you can specify the location of SSRPM server the port. It also enables/disables the appropriate authentication methods.

After the installation you will still need to configure site to use SSL as well as any other settings you wish to change (such as the CAPTCHA).

2.2.3. Manual Installation

This section will help you install IIS on the machine.

1. Go to 'Administrative Tools' and click on 'Server Manager'. This will open the following dialog:
2. Click on 'Add Roles and Features' to start the Add Roles and Features wizard.
3. Click on 'Next' to navigate to 'Server Roles'. Expand 'Web Server (IIS) and select the options 'ASP.NET 4.5' under 'Application Development'.
If you check 'ASP.NET 4.5' you will be shown the 'Add role services required for ASP.NET 4.5' dialog, because ASP.NET 4.5 requires that '.NET Extensibility 4.5' and 'ISAPI Extensions' are installed. Add both features to continue.
4. Expand 'Security' and select the following options: Basic Authentication, URL Authorization and Windows Authentication.
5. Click on 'Next' and check if ASP.NET 4.5 is checked.
6. Click on 'Next' and followed by a click on 'Install'.
7. On Windows 2008 it might be necessary to run the command "aspnet_regiis -i" in order to enable .NET4.5 functionality. This command should be run as Administrator.
Open the command prompt as Admin and execute the following command:
"C:\Windows\Microsoft.NET\Framework\v4.0.30319\aspnet_regiis -i".

2.2.4. Configuration

After installing both the SSRPM COM Object and the SSRPM Web Interface, IIS can now be configured to run the SSRPM Web Interface. This chapter describes in detail how to configure IIS to run the SSRPM Web Interface. There are multiple ways to set it up. In this example we setup two different web sites. Another option would be to setup a single web interface with two separate applications with different authentication methods.

Note: By default the web interface is configured for HTTP and is not as secure as it should be.

It is recommended to use HTTPS and to edit the web.config file and to add the highlighted attributes:

```
<httpCookies httpOnlyCookies="true" requireSSL="true" />
```

```
<forms loginUrl="~/Home/Index" defaultUrl="~/Home/Index" timeout="60" name=".SSRPMFORMSAUTH"  
domain="*" slidingExpiration="false" requireSSL="true"/>
```

It is also advisable to set the option "EnableHttpsRedirect" to true.

1. Copy the web interface files and folder to the appropriate directory, preferably something like:
"C:\inetpub\wwwroot\SSRPM_FormsAuth".
By default the source files of the web interface can be found in "C:\Program Files (x86)\Tools4ever\SSRPM\Admin Console\Examples\Client Web Interface".
2. Open the 'Internet Information Services (IIS) Manager'.
3. Right click on the server name and click on 'Add Website'. And enter the specified information.
It is advisable to use https and even mandatory if you want to use the web interface for the SSRPM desktop clients. If you already have a certificate please select 'https' and the correct certificate. If not, please change this to https after installing the certificate.
4. Select the new website and open the 'Authentication' dialog.
5. Make sure 'Forms Authentication' is enabled and 'Windows Authentication' is disabled.
6. Make sure the application pool used by the SSRPM web site is using .NET v4.
7. The IIS user needs to have access to the COM object DLL files. The quickest solution is to copy the COM object folder from "C:\Program Files (x86)\Tools4ever\SSRPM\" to "C:\inetpub\wwwroot\" and register the COM object from that location.
8. It is recommended to enable "Require SSL" in the SSL Settings.
9. The web interface with 'Forms Authentication' should now be running.
10. In order to setup the web interface with 'Windows Authentication' we repeat will the steps described earlier only with some minor differences. Please note that you only need the web interface with Windows authentication if you are going to use the enrollment wizard and you want the user to be automatically logged on.
11. Repeat step 1, but name the destination folder something like: "C:\inetpub\wwwroot\SSRPM_WinAuth".

12. Repeat step 3, but name the website 'SSRPM_WinAuth' and use the folder from step 11 as the physical path. Also change the port number so you can access both websites with the same URL, but different port numbers, e.g. `http://localhost:80` and `http://localhost:81`.
13. Do step 4 and 5, but instead of using 'Forms Authentication', disable 'Forms Authentication' and 'Anonymous authentication' and enable 'Windows Authentication'.
14. Now both web interfaces should be up and running.
15. If the web interface is running on a different machine as the SSRPM service or you aren't using the default port, please open the file 'web.config' of each web site and configure the correct settings. For more information about the available settings check the chapter 'Web.config'.

2.3. Configuring SSL

After the Web Interface has been installed, it can be used to enroll users and reset passwords. It is however not very safe, because all information is sent to the IIS Server in clear text. Because of this, all Web Interface Implementations should use SSL to encrypt all information sent and received. This chapter will guide you through the process of configuring SSL for the SSRPM Web Interface.

If you are using IIS 7 you can use the following steps to configure SSL.

1. Open the 'Internet Information Services (IIS) Manager'. The IIS Manager is displayed:
2. Right click on 'SSRPM Web Interface' and select 'Properties' from the menu.
3. Click on 'Bindings' in the right toolbar.
4. Click on 'Add' to create a new binding for https.
5. Click on 'Select...' to select one of the available certificates and click OK. For the purpose of this document we created a self-signed certificate named 'SSRPM'. There are several ways to get a SSL certificate but this will not be discussed in this document.
6. Close the open dialogs by clicking 'OK'.
7. Open the SSL settings of the 'SSRPM Web Interface' in 'Internet Information Services (IIS) Manager'. Check the box 'Require SSL' and click on 'Apply'.

After completing these steps, the web interface is secured using SSL. To access the web interface 'https' instead of 'http' must be used.

2.4. Configuring the Web Interface in a DMZ

The SSRPM Web Interface together with the SSRPM COM Object can be configured to run in an DMZ / perimeter network. This chapter describes how to configure IIS to be able to use the Web Interface in a DMZ.

Note: There are many possible configurations for a DMZ/perimeter network. The solution in this chapter is based on a network, that consists of three zones: LAN, DMZ and Internet. The solution provided in this document is written for this type of network, but can easily be modified to be used in other configurations.

Note 2: This chapter assumes that the SSRPM web interface has been installed and configured on a machine in the DMZ.

Please note: A connection rule in the firewall must be created to allow RPC traffic from the SSRPM COM Object in the DMZ to the SSRPM Service in the LAN. By default the SSRPM Service uses port 37946 (TCP) to communicate with it's clients.

IIS configuration:

1. Create an account in the domain. This account may be a member of 'Domain Guests' only.
2. Create a local account with the same name and password on the machine in the DMZ running the IIS server. (which may also be a guest account)
3. Install the SSRPM Web Interface on the machine in the DMZ. A complete guide on how to install the web interface can be found at the beginning of this chapter.
4. Open the IIS Manager on the machine in the DMZ.
5. Click on the SSRPM website.
6. Click on 'Authentication', this will result in the following dialog:
 7. If necessary, enable 'Forms Authentication' by right clicking on 'Forms Authentication' and clicking on 'Enable'.
 8. If necessary, enable 'Anonymous Authentication' by right clicking on 'Anonymous Authentication' and clicking on 'Enable'.
 9. Right Click on 'Anonymous Authentication' and click on 'Edit'. It will open the 'Edit Anonymous Authentication Credentials' dialog:
 10. Enter the username and password of the account that was created in Step 1/2
 11. Click on 'ok'.
 12. Close the IIS Manager.

2.5. Troubleshooting

Persistent -29 error

In some environments it is necessary to create an additional local user on the machine running the SSRPM service. This local user needs to have the same user name and password as the local user on the DMZ (IIS) and the domain user. Before trying this we recommend double checking the configuration described above and verifying that all the required ports are open.

Please note that the local user approach doesn't work if the DMZ server is a member of a DMZ domain. One solution would be to move the server from the domain to a workgroup.

3. Web.config

The "web.config" file is an XML file containing various settings, with which you can configure the some of the behavior of the web site as well as some security settings.

In this chapter we will only discuss the settings related to the behaviour of the web interface.

The file is located in the root folder of the web interface. By default it's assumed that HTTPS will be used, if you want to support HTTP you need to remove the "requireSSL" attributes from the elements "httpCookies" and "authentication\forms".

3.1. Log

Under the tag "<log4net>" you can configure the log settings:

Log mask

Description: The level of information that should be logged.

Attribute name: level

Value type: String

Possible values:

- ALL
- DEBUG
- INFO
- WARN
- ERROR
- FATAL
- OFF

Example: <level value="WARN" />

File name

Description: Configure the file name and location.

By default the log file is stored in:

"c:\programdata\Tools4ever\SSRPM\SSRPMClientWebInterface\Log\Log.txt".

Attribute name: file

Value type: String

Possible values:

- A valid path.

Example: <file type="log4net.Util.PatternString"
value="%envFolderPath{CommonApplicationData}/Tools4ever/SSRPM/SSRPMClientWebInterface/Log/Log.txt"
/>

File size

Description: The maximum size of the log.

Attribute name: maximumFileSize

Value type: String

Possible values:

- A valid size specification, e.g. 10MB

Example: <maximumFileSize value="10MB" />

Maximum files

Description: Maximum number of log files.

Attribute name: maxSizeRollBackups

Value type: Number

Possible values:

- 0-99

Example: <maxSizeRollBackups value="3" />

3.2. SSRPM Website

Under the tag "<SSRPMClientWebInterface.Properties.Settings>" you can configure the behavior of the web interface:

AllowUnenroll

Description: Determine whether or not the user is allowed to unenrolling using the web interface.

Attribute name: AllowUnenroll

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="AllowUnenroll" serializeAs="String">
  <value>True</value>
</setting>
```

CaptchaCharacterSet

Description: The specified characters that can be used by the CAPTCHA generation algorithm.

Attribute name: CaptchaCharacterSet

Value type: String

Possible values:

- A string of characters

Example:

```
<setting name="CaptchaCharacterSet" serializeAs="String">
  <value>ABCDEFGHIJKLMNOPQRSTUVWXYZ123456798</value>
</setting>
```

CaptchaEnableCookieStorageProvider

Description: The method used by the CAPTCHA to store the captcha value. If enabled it will store the value in a cookie, otherwise it will be stored in the session. Use this value if the CAPTCHA is not being displayed properly.

Attribute name: CaptchaEnableCookieStorageProvider

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="CaptchaEnableCookieStorageProvider" serializeAs="String">
  <value>False</value>
</setting>
```

CaptchaEnableMath

Description: Enable this setting to use a math-based CAPTCHA instead of the regular CAPTCHA which shows random characters.

Attribute name: CaptchaEnableMath

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="CaptchaEnableMath" serializeAs="String">
  <value>False</value>
</setting>
```

CaptchaEnableIntelligent

Description: Enable this setting to use a intelligent CAPTCHA. The intelligent captcha is hidden by default. It uses the one or more of the listed policies to check if user is a person.

The available policies are:

- Fake input policy - See CaptchaEnableFakeInputIntelligencePolicy for more details
- Javascript policy - See CaptchaEnableJavaScriptIntelligencePolicy for more details
- Response time policy - See CaptchaEnableResponseTimeIntelligencePolicy for more details
-

If it decides that it is not a person it will show the CAPTCHA, which will need to be passed before the user can continue.

Attribute name: CaptchaEnableIntelligent

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="CaptchaEnableIntelligent" serializeAs="String">
  <value>True</value>
</setting>
```

CaptchaEnableFakeInputIntelligencePolicy

Description: This CAPTCHA policy adds a fake and hidden input field to the form. This field has to remain empty for the policy to pass.

Based on the assumption that internet bots tend to enter data in all possible fields.

Attribute name: CaptchaEnableFakeInputIntelligencePolicy

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="CaptchaEnableFakeInputIntelligencePolicy" serializeAs="String">
  <value>True</value>
</setting>
```

CaptchaEnableJavaScriptIntelligencePolicy

Description: This CAPTCHA policy checks if the user has javascript enabled. If it is disabled the user has to enter the CAPTCHA.

Attribute name: CaptchaEnableJavaScriptIntelligencePolicy

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="CaptchaEnableJavaScriptIntelligencePolicy" serializeAs="String">
  <value>True</value>
</setting>
```

CaptchaEnableResponseTimeIntelligencePolicy

Description: This CAPTCHA policy checks if the response time of the user meets a certain minimum time. If the user fills out the form to quickly it is assumed that it is a internet bot and it will need to enter the CAPTCHA.

The time out value is configurable using the setting: CaptchaResponseTimeIntelligenceSeconds

Attribute name: CaptchaEnableResponseTimeIntelligencePolicy

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="CaptchaEnableResponseTimeIntelligencePolicy" serializeAs="String">
  <value>True</value>
</setting>
```

CaptchaLength

Description: The number of characters shown by the CAPTCHA.

Attribute name: CaptchaLength

Value type: Number

Possible values:

- 0-9

Example: <setting name="CaptchaLength" serializeAs="String">
<value>5</value>
</setting>

CaptchaHeight

Description: The height of the CAPTCHA.

Attribute name: CaptchaHeight

Value type: Number

Possible values:

- 0-9999

Example: <setting name="CaptchaHeight" serializeAs="String">
<value>65</value>
</setting>

CaptchaWidth

Description: The width of the CAPTCHA.

Attribute name: CaptchaWidth

Value type: Number

Possible values:

- 0-9999

Example: <setting name="CaptchaWidth" serializeAs="String">
<value>153</value>
</setting>

CaptchaIgnoreCase

Description: This setting determines whether or not the CAPTCHA is case-sensitive.

Attribute name: CaptchaIgnoreCase

Value type: Boolean

Possible values:

- True

Example: <setting name="CaptchaIgnoreCase" serializeAs="String">
<value>True</value>
</setting>

CaptchaResponseTimeIntelligenceMilliseconds

Description: This setting determines the minimum number of milliseconds that there should be between the page load and the submission of the form, before the user is considered to be a person and not a internet bot.

Attribute name: CaptchaResponseTimeIntelligenceMilliseconds

Value type: Number

Possible values:

- 0-9000

Example: <setting name="CaptchaResponseTimeIntelligenceMilliseconds" serializeAs="String">
<value>500</value>
</setting>

DefaultDomain

Description: The default domain of the users.

Attribute name: DefaultDomain
Value type: String
Possible values:

- domain.local

Example:

```
<setting name="DefaultDomain" serializeAs="String">
  <value></value>
</setting>
```

DefaultLanguage

Description: The default domain of the users.

Attribute name: DefaultLanguage

Value type: String

Possible values:

- English: en-US
- Dutch: nl-NL
- German: de-DE
- Spanish: es-ES
- Chinese: zh-CN
- Czech: cs-CZ
- Finnish: fi-FI
- Hungarian: hu-HU
- Italian: it-IT
- Polish: pl-PL
- Portuguese: pt-PT
- Russian: ru-RU
- Japanese: ja-JP
- Thai: th-TH
- Slovak: sk-SK

Example:

```
<setting name="DefaultLanguage" serializeAs="String">
  <value>en-US</value>
</setting>
```

EnableCaptcha

Description: Enable or disable the CAPTCHA.

Attribute name: EnableCaptcha

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="EnableCaptcha" serializeAs="String">
  <value>True</value>
</setting>
```

HomeButtons

Description: Specify the order of the buttons in the home screen. Only buttons shown in this list are displayed and are displayed in the order in which they are listed.

Attribute name: HomeButtons

Value type: String

- Possible values:*
- Enroll: Show the option to enroll
 - Reset: Show the option "Forgot my password"
 - ChangePassword: Show the option "Change password"
 - Forgotmyusername: Show the option "Forgot my user names"
 - Adselfservice: Show the manage your Active Directory information
 - Onboarding: Show the option to onboard.

Example:

```
<setting name="HomeButtons" serializeAs="String">
  <value>enroll, reset, changepassword, adselfservice, onboarding, forgotmyusername</value>
</setting>
```

EnableChangePasswordOnError1907

Description: Allow users who need to change their password on next logon to enroll by allowing them to change their current password during the enrollment process.

Attribute name: EnableChangePasswordOnError1907

Value type: Boolean

- Possible values:*
- True
 - False

Example:

```
<setting name="EnableChangePasswordOnError1907" serializeAs="String">
  <value>True</value>
</setting>
```

EnableConfirmPassword

Description: Show a confirm password edit box when providing a new password during the password reset or during the change password process.

Attribute name: EnableConfirmPassword

Value type: Boolean

- Possible values:*
- True
 - False

Example:

```
<setting name="EnableConfirmPassword" serializeAs="String">
  <value>True</value>
</setting>
```

EnableDomainInputEnroll

Description: Allow the users to specify their domain in an edit box during enrollment.

Attribute name: EnableDomainInputEnroll

Value type: Boolean

- Possible values:*
- True
 - False

Example:

```
<setting name="EnableDomainInputEnroll" serializeAs="String">
  <value>True</value>
</setting>
```

EnableDomainInputReset

Description: Allow the users to specify their domain in an edit box during reset.

Attribute name: EnableDomainInputReset

Value type: Boolean

- Possible values:*
- True
 - False

Example:

```
<setting name="EnableDomainInputReset" serializeAs="String">
  <value>True</value>
</setting>
```

EnableDomainSelectionEnroll

Description: Allow the users to select their domain from a drop-down list during enrollment.

Attribute name: EnableDomainSelectionEnroll

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="EnableDomainSelectionEnroll" serializeAs="String">
  <value>True</value>
</setting>
```

EnableDomainSelectionReset

Description: Allow the users to select their domain from a drop-down list during reset.

Attribute name: EnableDomainSelectionReset

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="EnableDomainSelectionReset" serializeAs="String">
  <value>True</value>
</setting>
```

EnableShowAnswerEnroll

Description: If enabled, a button is shown next to answer input field during enrollment. This button can be used to show the answer that was entered, which is obscured by default.

Attribute name: EnableShowAnswerEnroll

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="EnableShowAnswerEnroll" serializeAs="String">
  <value>True</value>
</setting>
```

EnableShowAnswerReset

Description: If enabled, a button is shown next to answer input field during the password reset process. This button can be used to show the answer that was entered, which is obscured by default.

Attribute name: EnableShowAnswerReset

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="EnableShowAnswerReset" serializeAs="String">
  <value>True</value>
</setting>
```

EnableShowPassword

Description: If enabled, a button is shown next to password input field. This button can be used to show the password that was entered, which is obscured by default.

Attribute name: EnableShowPassword

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="EnableShowPassword" serializeAs="String">
  <value>True</value>
</setting>
```

EnforceDefaultDomain

Description: If enabled, the user is not allowed enter a domain and the default domain is used.

Attribute name: EnforceDefaultDomain

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="EnforceDefaultDomain" serializeAs="String">
  <value>True</value>
</setting>
```

EnforcePasswordComplexityRules

Description: Determines whether or not the user can submit a password that does not match the password requirements as they are shown to the user.

Attribute name: EnforcePasswordComplexityRules

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="EnforcePasswordComplexityRules" serializeAs="String">
  <value>True</value>
</setting>
```

IPv4Only

Description: Configure the COM object to only connect to the service using the IPv4 protocol.

Attribute name: IPv4Only

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="IPv4Only" serializeAs="String">
  <value>True</value>
</setting>
```

OkBtnRedirectUrl

Description: Determine to which URL the user is redirected when he/she clicks on the OK-button at the end of a wizard process. This only applies to the web interface and only the setting "EnableShowWizardAsOverlay" is set to False.

Attribute name: OkBtnRedirectUrl

Value type: String

Possible values:

- A valid web address, for example: <https://www.tools4ever.com>

Example:

```
<setting name="OkBtnRedirectUrl" serializeAs="String">
  <value>https://localhost</value>
</setting>
```

Port

Description: The port on which the SSRPM service listens, by default 37946.

Attribute name: Port

Value type: Number

Possible values:

- 37946

Example:

```
<setting name="Port" serializeAs="String">
  <value>37946</value>
</setting>
```

Server

Description: The address of the server on which the SSRPM service is running. Multiple services can be specified as a comma separated list (for fail-over purposes).

Attribute name: Server

Value type: String

Possible values:

- server01.domain.local
- server01
- 127.0.0.1
- server01, server02, server03

Example:

```
<setting name="Server" serializeAs="String">
  <value>localhost</value>
</setting>
```

EnableLanguageDetection

Description: Shows the website in a specific language of the web interface based on the language settings configured in the user's browser.

Attribute name: EnableLanguageDetection

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="EnableLanguageDetection" serializeAs="String">
  <value>True</value>
</setting>
```

CustomDomainSelectionList

Description: A comma-separated string listing the available domains. This list can be used as a filter or as the list of available domains, as shown in the domain drop-down list, depending on the setting "UseCustomDomainSelectionListAsFilter".

Attribute name: CustomDomainSelectionList

Value type: String

Possible values:

- Domain01,domain02

Example:

```
<setting name="CustomDomainSelectionList" serializeAs="String">
  <value>Domain01,domain02</value>
</setting>
```

UseCustomDomainSelectionListAsFilter

Description: This setting can be used in conjunction with the setting "CustomDomainSelectionList". The list of specified domains will be used as a filter and only domains that are available (found by the service) and on the list will be shown.

Attribute name: UseCustomDomainSelectionListAsFilter

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="UseCustomDomainSelectionListAsFilter" serializeAs="String">
  <value>False</value>
</setting>
```

EnableShowWizardAsOverlay

Description: If enabled web interface opens the wizard in an overlay which gives it more of a client application look-and-feel.

Attribute name: EnableShowWizardAsOverlay

Value type: Boolean

Possible values:

- True
- False

Example: <setting name="EnableShowWizardAsOverlay" serializeAs="String">
<value>True</value>
</setting>

EnableRandomHomeBackground

Description: If enabled web interface loads a new random background when the home page is reloaded. The background remains the same during an enrollment or reset process. It is possible to add, remove or replace the background images to the folder "/content/images". Please note that the file name should start with "Background" and adding and removing background images requires a restart of the web interface.

Attribute name: EnableRandomHomeBackground

Value type: Boolean

Possible values:

- True
- False

Example: <setting name="EnableRandomHomeBackground" serializeAs="String">
<value>True</value>
</setting>

EnableClearCredentialsOnLogonError

Description: If enabled web interface doesn't prefill the credentials if authentication failed and the logon page is reloaded.

Attribute name: EnableClearCredentialsOnLogonError

Value type: Boolean

Possible values:

- True
- False

Example: <setting name="EnableClearCredentialsOnLogonError" serializeAs="String">
<value>True</value>
</setting>

EnableCaptchaInBrowserClient

Description: If enabled CAPTCHA is also used in the browser client. The browser client is the desktop application that is used to show the enrollment- and reset wizard.

Attribute name: EnableCaptchaInBrowserClient

Value type: Boolean

Possible values:

- True
- False

Example: <setting name="EnableCaptchaInBrowserClient" serializeAs="String">
<value>False</value>
</setting>

AutocompleteAttributeValue

Description: The autocomplete attribute is added to a form to inform the browser that it should not prefill the edit field of the form with data based on previously entered data. This attribute is ignored by Chrome.

Attribute name: AutocompleteAttributeValue

Value type: String

Possible values:

- On
- Off

Example:

```
<setting name="AutocompleteAttributeValue" serializeAs="String">
  <value>off</value>
</setting>
```

ValidationExpressionEmailAddress

Description: This is a regular expression that is used to validate the e-mail address that the user inputs during enrollment.
Please note that the default requirements set by SSRPM still apply and that this is only intended as an additional requirement.

Attribute name: ValidationExpressionEmailAddress

Value type: String

Possible values:

- A valid regular expression, e.g. `^.*@.*\$.com$`

Example:

```
<setting name="ValidationExpressionEmailAddress" serializeAs="String">
  <value>^.*@.*\$.com$</value>
</setting>
```

ValidationExpressionPhoneNumber

Description: This is a regular expression that is used to validate the mobile phone number that the user inputs during enrollment.
Please note that the default requirements set by SSRPM still apply and that this is only intended as an additional requirement.

Attribute name: ValidationExpressionPhoneNumber

Value type: String

Possible values:

- A valid regular expression, e.g. `^[0-9]*$`

Example:

```
<setting name="ValidationExpressionPhoneNumber" serializeAs="String">
  <value>^[0-9]*$</value>
</setting>
```

EnableUseDomainAsUpnSuffix

Description: Enable this option if you want use the default domain or specified domain as an UPN suffix instead of a NT4 domain name.

Attribute name: EnableUseDomainAsUpnSuffix

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="EnableUseDomainAsUpnSuffix" serializeAs="String">
  <value>False</value>
</setting>
```

EnableFirstItemInQuestionSelectListIsEmpty

Description: Enable this option if you want to add an empty entry to the top of the question selection drop-down list that is used during the enrollment. The empty entry will be selected by default.

Attribute name: EnableFirstItemInQuestionSelectListIsEmpty

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="EnableFirstItemInQuestionSelectListIsEmpty" serializeAs="String">
  <value>False</value>
</setting>
```

EnableFAQ

Description: Use this option to display or hide the FAQ in SSPRM. This option is enabled by default.

Attribute name: EnableFAQ

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="EnableFAQ" serializeAs="String">
  <value>True</value>
</setting>
```

EnableCustomLogo

Description: Use this option to have the web interface also replace the default Tools4ever logo with an DIV container with the class "Logo". The default Tools4ever logo is loaded as in a IMG container and as such the logo can't be replaced using CSS. Use this option if you want to replace the Tools4ever logo with a client specific logo. This option is disabled by default.

Attribute name: EnableCustomLogo

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="EnableCustomLogo" serializeAs="String">
  <value>False</value>
</setting>
```

EnabledCustomCss

Description: Use this option to have the web interface also load the file "custom.css", which should be used for client specific styling . This option is disabled by default.

Attribute name: EnabledCustomCss

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="EnabledCustomCss" serializeAs="String">
  <value>True</value>
</setting>
```

EnableAutoSendValidationMessageOnFocusOut

Description: Use this option to change the behaviour of the advanced authentication enrollment with PIN. By default the web interface will send a PIN code the moment the focus leaves the input field of the email address or phone number and the input is considered valid. If you set this option to false, it will no longer send the PIN code automatically, but the user will have to click on the "Send PIN code"-link. By default this option is enabled.

Attribute name: EnableAutoSendValidationMessageOnFocusOut

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="EnableAutoSendValidationMessageOnFocusOut" serializeAs="String">
  <value>True</value>
</setting>
```

EnableLanguageSelection

Description: Use this option to allow end users to select a language for the graphical user interface.

Attribute name: EnableLanguageSelection

Value type: Boolean

Possible values:

- True
- False

Example: <setting name="EnableLanguageSelection" serializeAs="String">
 <value>True</value>
 </setting>

SelectionLanguages

Description: Use this option to specify which languages are available for end user selection.

Attribute name: SelectionLanguages

Value type: String

Possible values:

- english,dutch,german,french,spanish,chinese,czech,finnish,hungarian,italian,polish,portuguese,russian,thai,slovak

Example: <setting name="SelectionLanguages" serializeAs="String">
 <value>english,dutch,german,french,spanish,chinese,czech,finnish,hungarian,italian,polish,portuguese,russian,thai,slovak</value>
 </setting>

EnableAdSelfServiceWinAuth

Description: Use this option to have the web interface use Windows Authentication when opening the AD Self Service page.

Attribute name: EnableAdSelfServiceWinAuth

Value type: Boolean

Possible values:

- True
- False

Example: <setting name="EnableAdSelfServiceWinAuth" serializeAs="String">
 <value>True</value>
 </setting>

EnableHttpsRedirect

Description: Use this option to have the web interface redirect to HTTPS.

Attribute name: EnableHttpsRedirect

Value type: Boolean

Possible values:

- True
- False

Example: <setting name="EnableHttpsRedirect" serializeAs="String">
 <value>True</value>
 </setting>

EnableIPBlocking

Description: Use this option to have the web interface (temporarily) block an IP address after a specific number of failed login attempts.

Attribute name: EnableIPBlocking

Value type: Boolean

Possible values:

- True
- False

Example: <setting name="EnableIPBlocking" serializeAs="String">
 <value>False</value>
 </setting>

IPBlockMaxAllowedLoginAttempts

Description: The maximum number of allowed login attempts before the IP is address is (temporarily) blocked. This applies to the enrollment, change password and AD self service functionality.

Attribute name: IPBlockMaxAllowedLoginAttempts

Value type: Number

Possible values:

- 0-99

Example: <setting name="IPBlockMaxAllowedLoginAttempts" serializeAs="String">
 <value>4</value>
 </setting>

IPBlockMaxAllowedIdentifyAttempts

Description: The maximum number of allowed identify attempts before the IP is address is (temporarily) blocked. This applies to the forgot my password functionality.

Attribute name: IPBlockMaxAllowedIdentifyAttempts

Value type: Number

Possible values:

- 0-99

Example: <setting name="IPBlockMaxAllowedLoginAttempts" serializeAs="String">
 <value>4</value>
 </setting>

IPBlockMaxAllowedRequests

Description: The maximum number of allowed requests before the IP is address is (temporarily) blocked. This applies to all requests.

Attribute name: IPBlockMaxAllowedRequests

Value type: Number

Possible values:

- 0-99999

Example: <setting name="IPBlockMaxAllowedRequests" serializeAs="String">
 <value>100</value>
 </setting>

IPBlockCountValidityDuration

Description: The duration for with the counts for maximum allowed login attempts, identify attempts and requests are valid. If the duration expires the counters are reset to 0.

Attribute name: IPBlockCountValidityDuration

Value type: Number

Possible values:

- 0-99

Example: <setting name="IPBlockCountValidityDuration" serializeAs="String">
 <value>4</value>
 </setting>

IPBlockTimeOuts

Description: The comma separated list specifies the timeout penalties for repeated offenses. The values are specified in seconds and the order determines which penalty applies for which offense. The value -1 represents indefinitely. In the example below the user first offense results in a 30 second ban, the second 1 minutes, the third offense 3 minutes, the fourth offense 1 hours and the 5th offense means a permanent ban. If you only use temporary timeout values and the user exceeds the number of specified timeouts, the last timeout value from the list is applied.

Attribute name: IPBlockTimeOuts

Value type: String

Possible values:

- A list of numbers

Example: <setting name="IPBlockTimeOuts" serializeAs="String">
<value>30,60,180,3600,-1</value>
</setting>

IPBlockWhitelist

Description: A comma separated list that specifies which IP address are not subject to the IP blocking rules. It is possible to specify a complete IP address (IPv4 or IPv6) or to use the wildcard or to specify a range. Examples: 10.1.* and 10.1.1.1-100.

Attribute name: IPBlockWhitelist

Value type: String

Possible values:

- A list of IP addresses

Example: <setting name="IPBlockWhitelist" serializeAs="String">
<value>127.0.0.1,1.1.1.1, 10.1.*,10.2.100-255</value>
</setting>

IPBlockEnableNotification

Description: If enabled a notification e-mail is sent to the general notification e-mail address configured in the Admin Console.

Attribute name: IPBlockEnableNotification

Value type: Boolean

Possible values:

- True
- False

Example: <setting name="IPBlockEnableNotification" serializeAs="String">
<value>True</value>
</setting>

EnableFooter

Description: If enabled a text is displayed at the bottom of the page. The text variable name is "SSRPM_FOOTER" and is read from the language file, but it empty by default.

Attribute name: EnableFooter

Value type: Boolean

Possible values:

- True
- False

Example: <setting name="EnableFooter" serializeAs="String">
<value>True</value>
</setting>

EnableEnrollmentDuringOnboarding

Description: Use this option to force users to enroll during the onboarding procedure.

Attribute name: EnableEnrollmentDuringOnboarding

Value type: Boolean

Value type: Boolean
Possible values:

- True
- False

Example:

```
<setting name="EnableEnrollmentDuringOnboarding" serializeAs="String">
  <value>True</value>
</setting>
```

OnboardingToken

Description: The token that needs to be provided during the import of the onboarding data to validate the import.
Attribute name: OnboardingToken
Value type: String
Possible values:

- 7e30bebc-d17c-4833-98b6-d4c09e076b24

Example:

```
<setting name="OnboardingToken" serializeAs="String">
  <value>7e30bebc-d17c-4833-98b6-d4c09e076b24</value>
</setting>
```

ConnectionErrorCodes

Description: A list of errorcode that will trigger the web interface to switch to a different SSRPM server, if multiple servers are specified by the value 'Server'.
Attribute name: ConnectionErrorCodes
Value type: Comma separated list of numbers
Possible values:

- -29, -7, 12001,1715

Example:

```
<setting name="ConnectionErrorCodes" serializeAs="String">
  <value>-29, -7, 12001,1715</value>
</setting>
```

OnboardingDateFormat

Description: The format used by the datepicker component when selecting a date during enrollment.
Attribute name: OnboardingDateFormat
Value type: String
Possible values:

- A valid date format, e.g. dd/MM/yyyy

Example:

```
<setting name="OnboardingDateFormat" serializeAs="String">
  <value>dd/MM/yyyy</value>
</setting>
```

FailOverTimeout

Description: The time interval in seconds after which the SSRPM web interface should fall back to the original order of the configured servers.
Attribute name: FailOverTimeout
Value type: Number
Possible values:

- 0-9999999

Example:

```
<setting name="FailOverTimeout" serializeAs="String">
  <value>600</value>
</setting>
```

EnableOnboardingRequireAgreement

Description: Use this option to require the user to check checkbox before he/she can continue with the onboarding process.

Attribute name: EnableOnboardingRequireAgreement
Value type: Boolean
Possible values:

- True or False

Example:

```
<setting name="EnableOnboardingRequireAgreement" serializeAs="String">
  <value>True</value>
</setting>
```

EnableOnboardingShowAttributeValue

Description: If enabled, a button is shown next to answer input field during onboarding. This button can be used to show the answer that was entered, which is obscured by default.

Attribute name: EnableOnboardingShowAttributeValue
Value type: Boolean
Possible values:

- True or False

Example:

```
<setting name="EnableOnboardingShowAttributeValue" serializeAs="String">
  <value>True</value>
</setting>
```

EnableOnboardingHideAnswers

Description: If enabled, the input boxes during onboarding are password fields by default.

Attribute name: EnableOnboardingHideAnswers
Value type: Boolean
Possible values:

- True or False

Example:

```
<setting name="EnableOnboardingHideAnswers" serializeAs="String">
  <value>True</value>
</setting>
```

EnableForgotMyUserNameEmail

Description: Use this option to show an input field where the user can enter an e-mail address to retrieve his/her user name. If enabled in combination with 'EnableForgotMyUserNameSms', the user will be able to choose which option he wants to use.

Attribute name: EnableForgotMyUserNameEmail
Value type: Boolean
Possible values:

- True or False

Example:

```
<setting name="EnableForgotMyUserNameEmail" serializeAs="String">
  <value>True</value>
</setting>
```

EnableForgotMyUserNameSms

Description: Use this option to show an input field where the user can enter a phone number to retrieve his/her user name. If enabled in combination with 'EnableForgotMyUserNameEmail', the user will be able to choose which option he wants to use.

Attribute name: EnableForgotMyUserNameSms
Value type: Boolean
Possible values:

- True or False

Example:

```
<setting name="EnableForgotMyUserNameSms" serializeAs="String">
  <value>True</value>
</setting>
```

EnableForgotMyUserNameAlwaysSucceed

Description: Use this option to show an error message if the user's information could not be found or if another error occurs while sending the user name to the user. By default no error messages are shown since they could be used to confirm if the e-mail address/phone number is valid.

Attribute name: EnableForgotMyUserNameAlwaysSucceed

Value type: Boolean

Possible values:

- True or False

Example:

```
<setting name="EnableForgotMyUserNameAlwaysSucceed" serializeAs="String">
  <value>True</value>
</setting>
```

AdSelfServiceMinimalSearchLength

Attribute name: This option specifies the minimal length of the search term used to look up user's in the Active Directory when using the manager or assistant field of Active Directory Self Service.

Value type: Number

Possible values:

- A valid number

Example:

```
<setting name="AdSelfServiceMinimalSearchLength" serializeAs="String">
  <value>3</value>
</setting>
```

AdSelfServiceAvailableUserColumns

Description: Use this option to specify which columns are available for user selection grid when using the manager or assistant field of Active Directory Self Service.

Attribute name: AvailableCivUserColumns

Value type: String

Possible values:

- CanonicalName,AccountSID,sAMAccountName,EmailAddress,CommonName,DisplayName,SurName,GivenName,Name

Example:

```
<setting name="AdSelfServiceAvailableUserColumns" serializeAs="String">
  <value>DisplayName</value>
</setting>
```

AdSelfServiceShownUserColumns

Description: Use this option to specify which columns are displayed by default for user selection grid when using the manager or assistant field of Active Directory Self Service.

Attribute name: ShownCivUserColumns

Value type: String

Possible values:

- CanonicalName,AccountSID,sAMAccountName,EmailAddress,CommonName,DisplayName,SurName,GivenName,Name

Example:

```
<setting name="AdSelfServiceShownUserColumns" serializeAs="String">
  <value>DisplayName</value>
</setting>
```

EnableOnboardingDateEdit

Attribute name: This option specifies whether or not the datapicker used during onboarding is editable.

Value type: Boolean

Possible values:

- True or False

Example:

```
<setting name="EnableOnboardingDateEdit" serializeAs="String">
  <value>False</value>
</setting>
```

EnableOnboardingKendoDatePicker

Attribute name: This option specifies whether or not the kendo datapicker used during If false is specified a simple datepicker is used.

Value type: Boolean

Possible values:

- True or False

Example:

```
<setting name="EnableOnboardingKendoDatePicker" serializeAs="String">
  <value>True</value>
</setting>
```

OnboardingDatePickerDefaultYear

Attribute name: This option specifies the year value that is selected by default when using the simple datepicker.

Value type: Number

Possible values:

- A valid number

Example:

```
<setting name="OnboardingDatePickerDefaultYear" serializeAs="String">
  <value>1980</value>
</setting>
```

ForgotMyUserNamePhoneNumberValidationRegex

Attribute name: Use this option to specify the regular expression that is used to validate phone number when using the 'Forgot my username' functionality.

Value type: Number

Possible values:

- A valid number

Example:

```
<setting name="ForgotMyUserNamePhoneNumberValidationRegex" serializeAs="String">
  <value>^[0-9]{7,15}$</value>
</setting>
```

AntiForgerySuppressIdentityHeuristicChecks

Attribute name: Use this option to disable identity heuristic checks for the anti forgery token validation. This can fix issues associated with error messages like: 'The provided anti-forgery token was meant for user "", but the current user is "Domain\Username"'. Please note that changes to this option require a restart of the web application (i.e. the application pool) before taking effect.

Value type: Boolean

Possible values:

- True or False

Example:

```
<setting name="AntiForgerySuppressIdentityHeuristicChecks" serializeAs="String">
  <value>False</value>
</setting>
```

3.3. CIV Website

Under the tag "<SSRPMCIVWebInterface.Properties.Settings>" you can configure the behavior of the web interface:

CaptchaCharacterSet

Description: The specified characters that can be used by the CAPTCHA generation algorithm.

Attribute name: CaptchaCharacterSet

Value type: String

Possible values:

- A string of characters

Example:

```
<setting name="CaptchaCharacterSet" serializeAs="String">
  <value>ABCDEFGHIJKLMNOPQRSTUVWXYZ123456798</value>
</setting>
```

CaptchaEnableCookieStorageProvider

Description: The method used by the CAPTCHA to store the captcha value. If enabled it will store the value in a cookie, otherwise it will be stored in the session. Use this value if the CAPTCHA is not being displayed properly.

Attribute name: CaptchaEnableCookieStorageProvider

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="CaptchaEnableCookieStorageProvider" serializeAs="String">
  <value>False</value>
</setting>
```

CaptchaEnableMath

Description: Enable this setting to use a math-based CAPTCHA instead of the regular CAPTCHA which shows random characters.

Attribute name: CaptchaEnableMath

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="CaptchaEnableMath" serializeAs="String">
  <value>False</value>
</setting>
```

CaptchaEnableIntelligent

Description: Enable this setting to use an intelligent CAPTCHA. The intelligent captcha is hidden by default. It uses the one or more of the listed policies to check if user is a person.

The available policies are:

- Fake input policy - See CaptchaEnableFakeInputIntelligencePolicy for more details
- Javascript policy - See CaptchaEnableJavaScriptIntelligencePolicy for more details
- Response time policy - See CaptchaEnableResponseTimeIntelligencePolicy for more details

If it decides that it is not a person it will show the CAPTCHA, which will need to be passed before the user can continue.

Attribute name: CaptchaEnableIntelligent

Value type: Boolean

Possible values:

- True
- False

Example: `<setting name="CaptchaEnableIntelligent" serializeAs="String">
<value>True</value>
</setting>`

CaptchaEnableFakeInputIntelligencePolicy

Description: This CAPTCHA policy adds a fake and hidden input field to the form. This field has to remain empty for the policy to pass. Based on the assumption that internet bots tend to enter data in all possible fields.

Attribute name: CaptchaEnableFakeInputIntelligencePolicy

Value type: Boolean

Possible values:

- True
- False

Example: `<setting name="CaptchaEnableFakeInputIntelligencePolicy" serializeAs="String">
<value>True</value>
</setting>`

CaptchaEnableJavaScriptIntelligencePolicy

Description: This CAPTCHA policy checks if the user has javascript enabled. If it is disabled the user has to enter the CAPTCHA.

Attribute name: CaptchaEnableJavaScriptIntelligencePolicy

Value type: Boolean

Possible values:

- True
- False

Example: `<setting name="CaptchaEnableJavaScriptIntelligencePolicy" serializeAs="String">
<value>True</value>
</setting>`

CaptchaEnableResponseTimeIntelligencePolicy

Description: This CAPTCHA policy checks if the response time of the user meets a certain minimum time. If the user fills out the form too quickly it is assumed that it is an internet bot and it will need to enter the CAPTCHA.

The time out value is configurable using the setting: CaptchaResponseTimeIntelligenceSeconds
Attribute name: CaptchaEnableResponseTimeIntelligencePolicy

Value type: Boolean

Possible values:

- True
- False

Example: `<setting name="CaptchaEnableResponseTimeIntelligencePolicy" serializeAs="String">
<value>True</value>
</setting>`

CaptchaLength

Description: The number of characters shown by the CAPTCHA.

Attribute name: CaptchaLength

Value type: Number

Possible values:

- 0-9

Example: `<setting name="CaptchaLength" serializeAs="String">
<value>5</value>
</setting>`

CaptchaHeight

Description: The height of the CAPTCHA.

Attribute name: CaptchaHeight

Value type:

Value type: Number
Possible values:
▪ 0-9999
Example: <setting name="CaptchaHeight" serializeAs="String">
<value>65</value>
</setting>

CaptchaWidth

Description: The width of the CAPTCHA.
Attribute name: CaptchaWidth
Value type: Number
Possible values:
▪ 0-9999
Example: <setting name="CaptchaWidth" serializeAs="String">
<value>153</value>
</setting>

CaptchaIgnoreCase

Description: This setting determines whether or not the CAPTCHA is case-sensitive.
Attribute name: CaptchaIgnoreCase
Value type: Boolean
Possible values:
▪ True
▪ False
Example: <setting name="CaptchaIgnoreCase" serializeAs="String">
<value>True</value>
</setting>

CaptchaResponseTimeIntelligenceMilliseconds

Description: This setting determines the minimum number of milliseconds that there should be between the page load and the submission of the form, before the user is considered to be a person and not a internet bot.
Attribute name: CaptchaResponseTimeIntelligenceMilliseconds
Value type: Number
Possible values:
▪ 0-9000
Example: <setting name="CaptchaResponseTimeIntelligenceMilliseconds" serializeAs="String">
<value>500</value>
</setting>

DefaultDomain

Description: The default domain of the users.
Attribute name: DefaultDomain
Value type: String
Possible values:
▪ domain.local
Example: <setting name="DefaultDomain" serializeAs="String">
<value></value>
</setting>

DefaultLanguage

Description: The default domain of the users.
Attribute name: DefaultLanguage
Value type: String

- Possible values:*
- English: en-US
 - Dutch: nl-NL
 - German: de-DE
 - Spanish: es-ES
 - Chinese: zh-CN
 - Czech: cs-CZ
 - Finnish: fi-FI
 - Hungarian: hu-HU
 - Italian: it-IT
 - Polish: pl-PL
 - Portuguese: pt-PT
 - Russian: ru-RU
 - Japanese: ja-JP

Example: `<setting name="DefaultLanguage" serializeAs="String">
<value>en-US</value>
</setting>`

EnableCaptcha

Description: Enable or disable the CAPTCHA.

Attribute name: EnableCaptcha

Value type: Boolean

- Possible values:*
- True
 - False

Example: `<setting name="EnableCaptcha" serializeAs="String">
<value>True</value>
</setting>`

IPv4Only

Description: Configure the COM object to only connect to the service using the IPv4 protocol.

Attribute name: IPv4Only

Value type: Boolean

- Possible values:*
- True
 - False

Example: `<setting name="IPv4Only" serializeAs="String">
<value>True</value>
</setting>`

Port

Description: The port on which the SSRPM service listens, by default 37946.

Attribute name: Port

Value type: Number

- Possible values:*
- 37946

Example: `<setting name="Port" serializeAs="String">
<value>37946</value>
</setting>`

Server

Description: The adres of the server on which the SSRPM service is running.

Attribute name: Server

Value type: String

Possible values:

- server01.domain.local
- server01
- 127.0.0.1

Example:

```
<setting name="Server" serializeAs="String">
  <value>localhost</value>
</setting>
```

EnableLanguageDetection

Description: Shows the website in a specific language of the web interface based on the language settings configured in the user's browser.

Attribute name: EnableLanguageDetection

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="EnableLanguageDetection" serializeAs="String">
  <value>True</value>
</setting>
```

AutocompleteAttributeValue

Description: The autocomplete attribute is added to a form to inform the browser that it should not prefill the edit field of the form with data based on previously entered data. This attribute is ignored by Chrome.

Attribute name: AutocompleteAttributeValue

Value type: String

Possible values:

- On
- Off

Example:

```
<setting name="AutocompleteAttributeValue" serializeAs="String">
  <value>off</value>
</setting>
```

EnableFAQ

Description: Use this option to display or hide the FAQ in SSPRM. This option is enabled by default.

Attribute name: EnableFAQ

Value type: Boolean

Possible values:

- True
- False

Example:

```
<setting name="EnableFAQ" serializeAs="String">
  <value>True</value>
</setting>
```

EnableCustomLogo

Description: Use this option to have the web interface also replace the default Tools4ever logo with an DIV container with the class "Logo". The default Tools4ever logo is loaded as in a IMG container and as such the logo can't be replaced using CSS. Use this option if you want to replace the Tools4ever logo with a client specific logo. This option is disabled by default.

Attribute name: EnableCustomLogo

Value type: Boolean

Possible values:

- True
- False

Example: `<setting name="EnableCustomLogo" serializeAs="String">
<value>False</value>
</setting>`

EnabledCustomCss

Description: Use this option to have the web interface also load the file "custom.css", which should be used for client specific styling . This option is disabled by default.

Attribute name: EnabledCustomCss

Value type: Boolean

Possible values:

- True
- False

Example: `<setting name="EnabledCustomCss" serializeAs="String">
<value>True</value>
</setting>`

EnableLanguageSelection

Description: Use this option to allow end users to select a language for the graphical user interface.

Attribute name: EnableLanguageSelection

Value type: Boolean

Possible values:

- True
- False

Example: `<setting name="EnableLanguageSelection" serializeAs="String">
<value>True</value>
</setting>`

SelectionLanguages

Description: Use this option to specify which languages are available for end user selection.

Attribute name: SelectionLanguages

Value type: String

Possible values:

- english,dutch,german,french,spanish,chinese,czech,finnish,hungarian,italian,polish,portuguese,russian

Example: `<setting name="SelectionLanguages" serializeAs="String">
<value>english,dutch,german,french,spanish,chinese,czech,finnish,hungarian,italian,polish,portuguese,russian</value>
</setting>`

MaxAnswerLength

Description: The default number of asterisks shown.

Attribute name: MaxAnswerLength

Value type: Number

Possible values:

- 15

Example: `<setting name="MaxAnswerLength" serializeAs="String">
<value>15</value>
</setting>`

AvailableCivUserColumns

Description: Use this option to specify which columns are shown in user selection grid.

Attribute name: AvailableCivUserColumns

Value type: String

Possible values:

- CanonicalName,sAMAccountName,InternalEmailAddress,EmailAddress,MobilePhoneNumber,AD_CommonName,AD_D

Example: `<setting name="AvailableCivUserColumns" serializeAs="String">
 <value>CanonicalName,sAMAccountName,AD_SurName,AD_GivenName</value>
 </setting>`

ShownCivUserColumns

Description: Use this option to specify which columns are displayed by default for user selection grid.

Attribute ShownCivUserColumns

name:

Value type: String

Possible values:

- CanonicalName,sAMAccountName,InternalEmailAddress,EmailAddress,MobilePhoneNumber,AD_CommonName,AD_D

Example: `<setting name="ShownCivUserColumns" serializeAs="String">
 <value>CanonicalName,AD_SurName,AD_GivenName</value>
 </setting>`

EnableHttpsRedirect

Description: Use this option to have the web interface redirect to HTTPS.

Attribute EnableHttpsRedirect

name:

Value type: Boolean

Possible values:

- True
- False

Example: `<setting name="EnableHttpsRedirect" serializeAs="String">
 <value>True</value>
 </setting>`

ConnectionErrorCodes

Description: A list of errorcode that will trigger the web interface to switch to a different SSRPM server, if multiple servers are specified by the value 'Server'.

Attribute ConnectionErrorCodes

name:

Value type: Comma separated list of numbers

Possible values:

- -29, -7, 12001,1715

Example: `<setting name="ConnectionErrorCodes" serializeAs="String">
 <value>-29, -7, 12001,1715</value>
 </setting>`

FailOverTimeout

Description: The time interval in seconds after which the SSRPM web interface should fall back to the original order of the configured servers.

Attribute FailOverTimeout

name:

Value type: Number

Possible values:

- 0-9999999

Example: `<setting name="FailOverTimeout" serializeAs="String">
 <value>600</value>
 </setting>`

4. Web Interface Overview

4.1. Main Window

The main window is displayed if the user browses to the Web Interface.

The user is presented with three options which are described in the following chapters:

- Enroll - Allows a user to enroll into the SSRPM Program.
- Forgot my password - Allows a user to reset his password (and/or optionally to unlock his account).
- Change password - Allows a user to change his password if he knows his current password.
- Manage your information in Active Directory - Allows a user to manage his information in Active Directory if enabled and properly configured.
- Onboarding - Allow a user to change his password and activate his account as well as to enroll into SSRPM based on data provided by the organization. By default this option is hidden.
-

4.2. Enrollment

If a user has chosen to enroll into SSRPM, he is first asked to logon:

If the user is already enrolled, he will be asked if he wants to re-enroll or to unenroll.

Should the user choose the un-enroll option, a pop up dialog will be shown to ask the user to confirm. If the user confirms the user will be unenrolled from SSRPM and will return to the main window. If the user cancels the user will return to the re-enroll/un-enroll web page.

If the user is not enrolled this page will not be displayed and the user will be presented with the questions as shown below.

Once the user is logged on and if advanced authentication is enabled with 2-factor authentication the user will be able to enter his e-mail address and/or phone number.

After that he will be asked to answer the questions specified in the profile:

After answering the required questions, the user can enroll by clicking the 'Next' button. If the enrollment succeeds, he will be presented with the following page:

After the enrollment process, the user can reset his password or optionally unlock his account.

4.3. Forgot my password

If a user has chosen to reset his password, he is first asked to identify himself:

After the user has identified himself, he is asked to answer the question he answered during enrollment:

After the user has answered the questions, he is asked to enter the PIN code that sent by e-mail (in this example):

After answering the questions the user can enter the new password:

After answering the required questions and entering a new password, the user can reset his password by clicking on the 'Next' button. If the user successfully resets his password, he will be presented with the following page:

4.4. Change password

If a user has chosen to change his password using SSRPM, he is first asked to logon:

After logging on the user can enter a new password:

If the user successfully changes his password, he will be presented with the following page:

4.5. Forgot my username

If a user has forgotten his/her username he can look it up using an email address or phone number (depending on the configuration).

After clicking next the username is sent to the user using the selected medium.

4.6. Manage your information in Active Directory

If a user has elected to change their information in Active Directory using SSRPM, he is first asked to logon:

After logging on the user can change his information:

If the user successfully changes his information he will be presented with the following page:

4.7. Onboarding

The onboarding procedure is intended to allow users who do not know their Active Directory account information to enroll into SSRPM and to retrieve their Active Directory account information. Please note this option is currently hidden by default.

The organization provides the data with which SSRPM can identify and authenticate the users. Using the provided data SSRPM can activate the user's account and allow the user to change his/her password as well as enroll the user into SSRPM.

The web interface has an interface which should be used to import the onboarding data, see the section *Import* for more information.

Identification

Each user has a single attribute that is used as an identifier. After identification the user needs to authenticate him/herself.

Authentication

The users can authenticate themselves by entering specific information or by entering a PIN code that was sent by either SMS or e-mail. What information the user needs to enter, depends on which information was provided by the organization.

When using SMS or e-mail authentication make sure that those options are enabled in the SSRPM Profile of the user. If you don't want to use SMS or e-mail authentication for SSRPM resets, but only for the onboarding process, it is possible to disable the functionality for reset by disabled it for the specific scenario using the SMS/e-mail configuration settings in the reset tab.

Onboarding

As part of the onboarding procedure the account of the user is enabled (if disabled) and a random password is set. The random password is set so that the user can be enrolled into SSRPM. After enrollment the user is allowed to specify his own password.

After successfully onboarding the user is shown his Active Directory account information, consisting of the SAM account name, the domain name and the user principal name.

Security

The database stores user objects with attributes that can be used for identification or authentication.

These attributes are usually hashed, except for the e-mail address and phone number attributes, these are encrypted since we need to be able to get the original value in order to send a PIN code.

After a user successfully finished the onboarding procedure the attributes are removed from the database and the user's status is set to 'processed'.

Windows Credential Provider

It's possible to use this functionality from the Window logon screen. This requires the clients to be installed and properly configured. For more information about the available client configuration options, please refer to the GPO Distribution Guide.

4.7.1. Import

The SSRPM web interface has an interface which can be used to import the onboarding data.

In order to import the onboarding data you need to POST JSON to a URL that looks like:
<https://YOURSSRPMWEBINTERFACEURL/onboarding/import>

The interface supports the actions: "new" and "delete".

The value of the attribute with the option 1 is used to identify the user. If a user with that ID exists it is replaced or deleted, depending on the selected action.

The following data should be provided:

- Action : new or delete
- OnboardingToken: The provided onboardingToken should match the token specified in web.config file.
- Users: An array of user objects.

A user object has the following attributes:

- AccountSID: This value is optional. If this is missing the SSRPM service will look it up.
- Domain: The NETBIOS name of the domain.
- SAMAccountName: The SAM account name of the user.
- OnboardingDate: This value is optional. This date represents the date after which a user can start the onboarding process.
- Attributes: An array with attribute objects.

An attribute object has the following attributes:

- Name: Name of the attribute.
- Value: Value of the attribute.
- Options: Mask value that indicates the type of the attribute.

Available mask values:

1: Identifier attribute (must be unique).

2: A data validator attribute.

4: A SMS validator the value should be a mobile phone number.

8: An e-mail validator the value should be an e-mail address.

16: Indicates that the attribute value should be checked case sensitive.

32: Indicates that the attribute value is a date. The format should be: "DD/MM/YYYY".

For example: 1 means it's the users unique identifier (used to logon)

2 is for the "question" attributes and 18 (2+16) means that it is case sensitive.

Example (JSON - new):

```

{
Action: "new",
OnboardingToken: "7e30bebc-d17c-4833-98b6-d4c09e076b24",
users: [
  {
    Domain: "CATS",
    SAMAccountName: "jsmith",
    OnboardingDate: "2016-12-21",
    Attributes: [
      {
        Name:"ID",
        Value: "1234",
        Options:"1"
      },
      {
        Name:"ZIPCODE",
        Value: "1234AB",
        Options:2
      },
      {
        Name:"BSN",
        Value: "ABCDEF",
        Options:2
      },
      {
        Name:"email",
        Value: "jsmith@company.com",
        Options:8
      },
      {
        Name:"SMS",
        Value: "00123456789",
        Options:4
      }
    ]
  },
  {
    Domain: "CATS",
    SAMAccountName: "twilliams",
    AccountSID:"1-1-1-1-1111"
    Attributes: [
      {
        Name:"ID",
        Value: "12",
        Options:1
      },
      {
        Name:"Maidenname",
        Value: "Jones",
        Options:2
      },
      {
        Name:"email",
        Value: "twilliams@company.com",
        Options:8
      }
    ]
  }
]
}

```

Example (JSON - delete):

```
{
  Action: "delete",
  OnboardingToken: "7e30bebc-d17c-4833-98b6-d4c09e076b24",
  users: [
    {
      Domain: "",
      SAMAccountName: "",
      Attributes: [
        {
          Name:"ID",
          Value: "12",
          Options:1
        }
      ]
    }
  ]
}
```

4.7.2. Language

The onboarding data doesn't only supports a single attribute name per attribute. In order to be able to support multiple translations. SSRPM converts the name of an attribute to a text variable. It will look for the variable in the language files (located in the folder: c:\inetpub\SSRPM\SSRPM_FA\App_Data\Languages). The variables have the following format:

ONBOARDING_ATTRIBUTE_LABEL_ *ATTRIBUTENAME* = Translation of attribute name:
 ONBOARDING_ATTRIBUTE_PH_ *ATTRIBUTENAME* = Enter the value of your attribute.
 ONBOARDING_ATTRIBUTE_VALIDATION_ERROR_MESSAGE_ *ATTRIBUTENAME* = The value of attribute X is incorrect.

For example, the translations for the texts associated with the attribute "zipcode" would look like:
 ONBOARDING_ATTRIBUTE_LABEL_ZIPCODE = Your zip code:
 ONBOARDING_ATTRIBUTE_PH_ZIPCODE = Enter your zip code.
 ONBOARDING_ATTRIBUTE_VALIDATION_ERROR_MESSAGE_ZIPCODE = The specified zip code is incorrect.

4.7.3. Wizard

If a user has to onboard using SSRPM, he is first asked to identify himself:

After the user has identified himself, he is asked to select a validation method.
 Possible options are entering personal data, a PIN sent by SMS or a PIN sent by e-mail.

If the user decides to enter personal data in order to validate himself he will presented a screen like:
 The validation data is provided by the organization and the translated texts need to be added to the language files of the web interface.

After this screen the user is (by default) redirected to the enrollment procedure. See the *Enrollment* for more information about the enrollment procedure.
 After the enrollment the user needs to change his password.

After successfully changing his password the user is presented the success page.
 This page contains the information the user needs to login to the Windows environment.

5. Frequently Asked Questions (FAQ)

Can I change anything in the Web Interface?

No, but you can customize the look and feel. It is even possible to write your own web interface. Please refer to the "COM Object Guide" for a complete list of features.

Is the web interface secure?

It depends on which type of connection you are using. If you are using a default http connection, the connection is not secure. However, the connection can be secured using SSL. Please refer to the 'Installing the Web Interface' chapter in this document for a detailed description how to configure SSL.

Is the SSRPM COM Object secure?

Yes. The SSRPM COM Object communicates with the SSRPM Service using an encrypted connection.

I am trying to Enroll/Reset a password/Unlock Account and am getting error What is the problem?

Please refer to the "Implementation Guide" for a list of commonly encountered errors.

Wizard • 36

6. Index

C

Change password • 32
CIV Website • 25
Configuration • 4
Configuring SSL • 5
Configuring the Web Interface in a DMZ • 6

E

Enrollment • 31, 36

F

Forgot my password • 32
Forgot my username • 32
Frequently Asked Questions (FAQ) • 37

I

Import • 33, 34
Installing the SSRPM COM Object • 1
Installing the Web Interface • 1, 2
Introduction • 1

L

Language • 36
Log • 8

M

Main Window • 31
Manage your information in Active Directory • 32
Manual Installation • 2, 3
MSI Installation • 2, 3

O

Onboarding • 33

R

Requirements • 2

S

SSRPM Website • 9

T

Troubleshooting • 7

W

Web Interface Overview • 31
Web.config • 7