Test procedures for LSS

Table of contents

Version history

2014	Version 0.9	TSS
2014	Version 0.1	MSP

1 The purpose and audience of the document

This document lists the test procedures that must be followed and reported by a LSS vendor implementing and integrating with the LSS for NemID API.

The document is aimed at the testers of a LSS for NemID implementation and integration setup.

Summary of all documents in the LSS for NemID Package:

Implementation documentation

- Technical specification for LSS
- Implementation guide for LSS

Test documentation

- Guidelines on the use of LSS for NemID test tools
- Test procedures for LSS

Solution documentation

- Requirements feedback form for LSS
- End-customer documentation for LSS

2 Introduction

In order for a Local Signature Server (LSS) supplier to be able to offer LSS for NemID for its users, the supplier has to implement, setup and maintain a LSS for NemID backend on the users' local network.

This document describes the test procedure for a LSS supplier integrating to the LSS for NemID setup.

The LSS supplier's tests are divided into three parts. The first part is a functional test with purpose to verify whether the integration has been completed successfully and works as intended. The second part is test focused on a set of user-scenarios testing parts of the integration. It is a requirement for the second part that the first functional tests have been completed successfully.

In the last part of the tests the bootstrap setup is tested. Prerequisites for these tests is that the DNS for https://nemlogincss.dk is working on the end-user's local network and that the trust for the (custom) SSL-certificate has been set up on end user devices.

3 Test steps

The service provider package has a fully working example of an example service provider page, containing a demonstration of the integration to NemID Private, NemID Business and LSS for NemID. The URL used for the iFrame is configurable in the web application configuration. This can be used to setup a working service provider example integrating to the LSS supplier's iFrame while testing the setup. Refer to the documentation for the service provider demo web application (the LSS for NemID service provider example source) for more details on this.

The recommended test steps are outlined as follows:

1. Setup an example service provider page using the service provider package.

2. Use the reference implementation of a LSS to make the flow work from end-to-end.

3. Switch the LSS backend from the reference implementation to your LSS backend implementation.

4. Optional, the service provider example web application contains a test page tailored to test the response from a LSS backend iFrame. This can be used to test the integration.

5. Complete the tests described in this document.

6. Test the production setup on the network. This includes DNS to the bootstrap address and SSL-trust on end user devices.

4 Functional tests

This section describes the test procedure to verify the technical parts of the NemID integration.

4.1 Test of HTML5 Web Messaging setup

The communication between the LSS backend and the service provider, through the iFrame hosting the LSS backend web entry point, is handled in JavaScript and by using the HTML5 Web Messaging standard. The LSS supplier must test and verify that the JavaScript API function "`Parameters`" has been setup and is working, and that the iFrame entry point is sending the "`SendParameters`" command when the page is loaded and ready to receive the "`Parameters`" command.

In addition the LSS supplier tests that the backend is able to parse the JSON packages received.

4.2 Handling the /<random digits> in the request

The service provider will request

https://nemlogincss.dk/<random-digists>

Test that the web server handling the requests are able to handle this. The random digits should just be ignored.

4.3 Identify erroneous JSON

Return the correct error code to the service provider when the commands received contains improperly constructed JSON structures.

4.4 Missing parameter error code

Return the appropriate error code to the service provider when a required parameter is missing.

As an example, when signing a document a sign text parameter is required.

4.5 Parameter case-insensitivity

The parameters received in the JSON structure are mapped in name-value pairs. The names of the values are case-insensitive. Test that this is handled correctly.

4.6 Validate Timestamp

One of the parameters received from the service provider is a timestamp. Respond correctly on invalid values.

4.7 Validate Signature on parameters

The service provider provides a signature on the parameters sent to the LSS. The LSS backend should validate the signature.

Test and verify that your setup is able to validate the signature.

4.8 Validate Digest over parameters

The LSS backend must be able to calculate the exact same digest over the received parameters as the one generated by the service provider.

4.9 Error codes

Test that the correct error codes are returned for all specific error codes.

To this end setup and test scenarios for all error codes specified in the LSS technical specification documentation.

The **LSSGLB001** error code scenario is not tested. This should never be returned by the LSS supplier.

4.10 Test of XML-DSig returned to the service provider

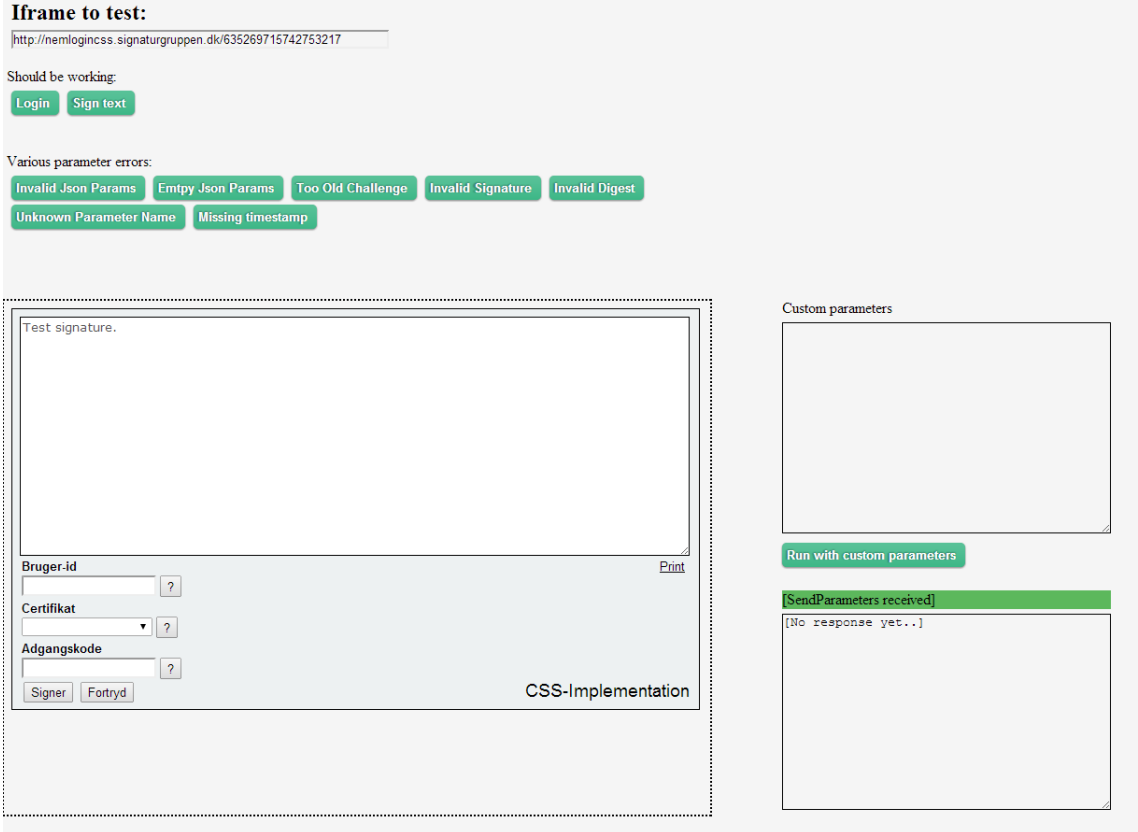
Use OOAPI to verify that the XML-DSig documents returned to the service provider validates. The service provider package example demo web application can be used as a reference.

5 Test integration page

The service provider package example demo web application includes a test page tailored for integration testing various scenarios of the integration.

The .Net example can be found in /testing/TestFrameIntegration.aspx.

It is illustrated below.



The web example has to be setup and working with a valid test-certificate to sign the parameters.

The page generates valid parameters for login and signing a simple text, which can be used as a reference. The "login" and "Sign text" buttons demonstrate these two flows.

The page is trying to communicate through the API setting up an iFrame using the URL specified in the "Iframe to test:" text-field. Use this to point at your own end-point.

There are buttons for various flows which are expected to fail. The text area in the right bottom area shows the result returned to the service provider and is lit up green when the return code is as expected.

A text-area for testing with custom parameters is available.

A "[SendParameters received" is lit up if the test page successfully receives the "SendParameters" command from the iFrame.

6 Use case tests

Complete the same use cases described in "SP Recommended test procedures". The same outcome is expected for all cases.

7 Sign text validation and rendering

[Not resolved yet. Current documentation on this is subject to change]

8 Bootstrap test

In order for a user to use LSS for NemID he or she must be connected to a local network with local DNS for the bootstrap address https://nemlogincss.dk.

Test that DNS is setup and working as intended.

The setup requires SSL. So the LSS backend iFrame must be hosted using custom SSL certificates for nemlogincss.dk. Trust for this certificate must be setup on the users' devices.