# Platform Tests

## Revision History

| Version | Date | By | Comments |
|---|---|---|---|
| 1 | 06/01/10 | Brian Graversen | Initial release of document |
| | | | |
| | | | |

# Introduction

As part of the SignOn Library delivery, the library has been tested on the following platforms, using a clean-installed OS for each of the platforms, installing only the necessary frameworks (.NET and Java if needed), and updating the OS through Windows Update to the latest service packs.

In general we test the OS with the latest OS servicepack, and with then with the minimum supported .NET framework (2.0 SP1 for Windows 2000/XP, and 3.5SP1 for Vista/Win7) and the latest (currently 3.5 SP1, except for Windows 2000, which only supports .NET 2.0 SP1).

**Platforms Tested**
[X] Windows 2000 Pro SP4 with .NET Framework 2.0 SP1
[X] Windows XP Pro SP3 with .NET Framework 2.0 SP1
[X] Windows XP Pro SP3 with .NET Framework 3.5 SP1
[X] Windows Vista SP1 with .NET Framework 3.5 SP1
[X] Windows 7 with .NET Framework 3.5 SP1

**Notes regarding .NET Framework SP1**
The SignOn Library itself works fine with .NET Framework 2.0 (without SP1), but certain functionality in CAPI is broken in that version, so to ensure that the CAPI PKI Plugin works, at least .NET Framework 2.0 SP1 is required.

It is always recommended to upgrade to the latest servicepack of .NET, to ensure any known security issues with the framework has been patched.

## What does the test cover?

The platform tests uses the sample code found bundled with the SignOn Library to build small client applications in both C# and Visual C (COM) to test the SignOn Library framework, and to test the GW Plugin and CAPI PKI Plugin.

For the test a SOSI GW has been installed and configured to run in test-mode, accessing the test-STS available on the internet. The client applications are configured to access this SOSI GW through the GW Plugin, and a single MOCES is installed in the machines CAPI keystore.

The small client applications configures the framework, the plugins and finally executes a single call to SignOn(), and the test is passed successfully if a IDCard Handle is returned, and an IDCard issued to the SOSI GW that matches this IDCard Handle. The platform test does not cover various error situations like wrong password, since these are all covered by the unittests.

All tests are performed using the default settings of the OS and with an ordinary user without any additional administration privileges.

## Goal / Success Criteria

The main goal of the platform tests are to ensure that the library will work on the tested platforms, without any additional dependencies having to be installed, and without any unexpected sideeffects.

When all of the above mentioned tests have passed, the SignOn Library is considered ready to use for these platforms (under the assumption that the other tests/unittests all pass as well).

## Notes regarding the executed tests

During the execution of the tests, the following observations where made.

**Windows 2000**

While the SignOn Library itself worked flawlessly with .NET 2.0, the CAPI PKI Plugin required .NET 2.0 SP1, otherwise the password would be rejected due to a bug in .NET that prevents passing the password programatically to the CSP; the password-dialog option worked fine, but this is not an option for the SignOn Library, which does not allow user-interaction during the signon process.

**Windows XP**

No unexpected behaviour observed while testing with either version of the .NET framework (2.0 SP1 and 3.5 SP1).

**Windows Vista**

The user under Windows Vista has less privileges than a corresponding user under Windows XP. This does not prevent the SignOn Library from working as expected, but if the framework attempts to write anything to the logfile, and the user does not have permissions to write to the logfile location no log-information will be written, which will make it difficult to track down any errors on that platform – vendors should ensure that any log-information written to disk, is written to a location that the user has permissions to access.

**Windows 7**

No unexpected behaviour observed while testing. As with Windows Vista, a default user may not have access rights to write logfiles anywhere, so in case a logfile is used by the SignOn Library, ensure that the user has the privileges needed to write to that location.