
# eIDAS-Node Migration Guide

Version 1.4.3

## Document history

| Version | Date | Modification reason | Modified by |
|---|---|---|---|
| 1.4 Pre release | 31/08/2017 | Origination | DIGIT |
| 1.4 Official release | 06/10/2017 | Minor update to section 3.7 | DIGIT |
| 1.4.1<br><br>Official release | 15/06/2018 | Document Version changed to correspond with the release. Reuse of document policy updated. | DIGIT |
| 1.4.2 | 06/08/2018 | Migration guide updated to describe changes made by removing vulnerability EID-631: Issuer URL in SAML AuthnRequest can be manipulated. | DIGIT (CB) |
| 1.4.3 | 11/09/2018 | Guide describing migration to eIDAS-Node v1.4.3 (from eIDAS-Node v1.4.2). | DIGIT |

**Disclaimer**

This document is for informational purposes only and the Commission cannot be held responsible for any use which may be made of the information contained therein. References to legal acts or documentation of the European Union (EU) cannot be perceived as amending legislation in force or other EU documentation.

The document contains a brief overview of technical nature and is not supplementing or amending terms and conditions of any procurement procedure; therefore, no compensation claim can be based on the contents of the present document.

## Table of contents



 

# 1. Introduction

This document is intended for a technical audience consisting of developers, administrators and those requiring detailed technical information on how to configure, build and deploy the eIDAS-Node application.

The purpose of this document is to facilitate migration from eIDAS-Node v1.4.2 to eIDAS-Node v1.4.3.

## 1.1. Document structure

This document is divided into the following sections:

Chapter 1 — *Introduction*: this section.

Chapter 2 — *Prerequisites*: Identifies any prerequisites that are required before migrating your eIDAS-Node to version 1.4.2.

Chapter 3 — *Changes*: Contains detailed information about the changes that should be taken into consideration when migrating to eIDAS-Node version 1.4.3.

## 1.2. Document aims

The main aim of this document is to provide information on all the changes requiring your action when migrating to eIDAS-Node version 1.4.3, including:

configuration changes; and

changes to common code.

## 2. Prerequisites

Before starting your migration to eIDAS-Node version 1.4.3 you should have:

already implemented eIDAS-Node version 1.4.2;
downloaded the eIDAS-Node v1.4.3 Integration Package; and
downloaded the latest documentation.

# 3. Changes

## 3.1. Summary of changes

In eIDAS-Node version 1.4.3 there are several changes that affect your installation. The main changes relate to resolving three EID issues:

EID-617 - Error responses contains assertions with a false identity

EID-630 - Missing Assertion in failed authentication response should be OK

EID-643 - Wrong character encoding in ConnectorMetadata

## 3.2. Allow reception and generation of not successful SAML Response without assertion

In previous versions, generated not successful SAML Responses that always contained an assertion. If the node received this not successful SAML Responses without an assertion, an exception was thrown and the flow stopped.

In the current version, this behaviour was changed and not successful SAML Responses do not include by default an assertion, anymore. However, a temporary property was added to eidas.xml, to allow sending of not successful SAML responses with an assertion, to maintain interoperability with older versions, including 1.4.2 and 2.1. For this, it uses the value of application identifier of protocol versioning, published in the requester's metadata.

The validation for not successful SAML Responses was relaxed, so that, not successful SAML Responses, with or without an assertion, are accepted.

### 3.2.1. Code changes

The code was changed so that failure responses with or without an assertion are accepted. AUCONNECTORSAML class was changed to cope with audience restriction being null for these cases. AUSERVICESAML was changed to call a new method in ProtocolEngineI which receives, as a parameter, a list of application identifiers to which responses must contain an assertion. Code was also extracted to methods generateResponseErrorMessage and getIncludeAssertionApplicationIdentifiers.

A new method was added to ProtocolProcessorI marshalErrorResponse that overloads the existing one and contains an additional parameter listing application identifiers of requester(s). The implementations of ProtocolProcessorI: EidasProtocolProcessor, EidasProtocolProcessorWithAutorization and StorkProtocolProcessor were also changed to implement the new method introduced. Similarly, the ProtocolEngineI was also changed and the method generateResponseErrorMessage was overloaded with an extra parameter applicationIdentifiers. The implementation of the new method contained in the interface was done at ProtocolEngine.

Method extractVerifiedAssertion in ResponseUtil was changed, so that, an assertion is only extracted in the case of successful responses and when only one assertion is in the response. Junit tests were added to cover this change at ResponseUtilTest. At EidasStringUtil, getTokens method was added to split a string into tokens separated by either semicolon or comma. Junit EidasStringUtilTest class was created to test this new

method. EidasParameterKeys was changed to contain the key for the new property added to eidas.xml: include.assertion.fail.response.application.identifiers.

#### 3.2.2. Configuration changes

A new property was introduced in the eidas.xml:

<entry key="include.assertion.fail.response.application.identifiers">CEF:eIDAS-ref:1.4.2;CEF:eIDAS-ref:2.1</entry>

The values of this property are protocol versioning's application identifiers, published in the requester's metadata, related to the case when not successful responses must include the assertion, due to interoperability reasons. For the other cases, the not successful responses will not contain an assertion.

### 3.3. Adapt Connector and Proxy Service Metadata to handle non-ASCII characters

When using non-ASCII characters in one of the metadata fields configurable through eidas.xml (eg. by specifying them in one of the *.contact.support.company entries) the servlet standard encoding of UTF-8 is now used for the generated XML metadata.

#### 3.3.1. Code changes

```
EIDAS-Node\src\main\java\eu\eidas\node\connector\ConnectorMetadataGeneratorServlet.java
EIDAS-Node\src\main\java\eu\eidas\node\service\ProxyServiceMetadataGeneratorServlet.java
```

#### 3.3.2. Configuration changes

No configuration changes made.

### 3.4. Other fixes/improvements

#### 3.3.1 Metadata issuer whitelist URL is case insensitive, should be sensitive

The issuer of either SAML Request or Response is tested against a whitelist of allowed URL's in case sensitive manner.