

eIDAS-Node Migration Guide v2.8

Table of Contents

1	Introduction	4
1.1	Document structure	4
1.2	Document aims.....	4
2	Prerequisites	5
3	Changes	6
3.1	Summary of EID Issues.....	6
3.2	Extraction of Logging classes to the Logging module	6
3.2.1	Code changes.....	6
3.3	Generify Loggers and Logging Filters	8
3.3.1	Code changes.....	8
3.4	Create a contract between the app for data used in JSP from the HttpServletRequest Request ...	9
3.4.1	Code changes.....	9
3.5	Extraction of Security classes to the Security module	10
3.5.1	Code changes.....	10
3.6	Introduction of OCSP / CRL Checking	11
3.6.1	Code changes.....	11
3.7	Metadata Derivatives Belong in Metadata not ProtocolProcessor	13
3.7.1	Code changes.....	13
3.8	Code cleanup	15
3.8.1	Removal of deprecated code	15
3.8.2	Deprecation of EidasNodeException.....	20
3.9	Removal of legacy keystore configuration.....	21
3.9.1	Code Changes	21
3.9.2	Configuration Changes	23
3.10	Support of 1.4 Technical Specifications.....	23
3.10.1	Only 1 Dataset can be Requested	23
3.10.2	Block Multi Dataset Request from Connectors 1.4 Spec and Higher	23
3.10.3	Removal of rsa-oaep-mgf1p URI	23
3.10.4	Block Responses with rsa-oaep-mgf1p URI coming from Proxies 1.4 Spec or higher	24
3.10.5	Update Default Protocol Version to Include 1.4	24
3.11	Upgrade of Bouncy Castle library to version 1.76	24
3.11.1	Code changes.....	24
3.12	Log Details of EIDASSAMLEngineException	25
3.12.1	Code changes.....	25
3.13	Sign with provider that loads the keys	26
3.13.1	Code changes.....	26
3.14	Other changes requiring no action.....	27
3.14.1	Defaults in SamlEngine.xml	27
3.14.2	Upgrade dependency of xmlsec for CVE-2023-44483.....	27
3.14.3	Reduce the declaration of namespaces in metadata XML	27
3.14.4	Reduce the declaration of http://www.w3.org/2001/XMLSchema-instance in Response Attributes 27	
3.14.5	SignModule_Connector_EC.xml and SignModule_Service_EC.xml config files do not contain metadata.signature.algorithm property	27
3.14.6	Replace SignatureConfiguration constructor by Builder.....	28
3.14.7	Binary light token hash algorithm and provider are now a property of LightRequest and LightResponse Loggers	28
3.14.8	Check Representative for MDS.....	28
3.14.9	Update the Application Identifier to 2.8.0	29
3.14.10	Set Logback.xml verbosity to Production	29
3.14.11	Create tests for missing entries in Error properties	29
3.14.12	Example HSM P12 example missing metadata.signature.algorithm for RSA	29
3.14.13	Remove password from configuration file	30

Document history

Version	Date	Modification reason	Modified by
1.0	04/09/2023	Created	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 information of a technical nature and does not supplement or amend the terms and conditions of any procurement procedure; therefore, no compensation claim can be based on the contents of this document.

© European Union, 2023

Reuse of this document is authorised provided the source is acknowledged. The Commission's reuse policy is implemented by Commission Decision 2011/833/EU of 12 December 2011 on the reuse of Commission documents.

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 v2.7 to eIDAS-Node v2.8.

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 2.8.

Chapter 3 — *Release Highlights* Chapter 4 — *Changes*: Contains detailed information about the changes that should be taken into consideration when migrating to eIDAS-Node version 2.8.

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 2.7, including:

- configuration changes; and
- changes to code.

Disclaimer: The users of the eIDAS-Node sample implementation remain fully responsible for its integration with back-end systems (Service Providers and Identity Providers), testing, deployment and operation. The support and maintenance of the sample implementation, as well as any other auxiliary services, are provided by the European Commission according to the terms defined in the European Union Public Licence (EUPL) at https://joinup.ec.europa.eu/sites/default/files/custom-page/attachment/eupl_v1.2_en.pdf

2 Prerequisites

Before starting your migration to eIDAS-Node version 2.8 you should have:

- Already implemented eIDAS-Node version 2.7
- Downloaded the eIDAS-Node v2.8 Integration Package; and
- Downloaded the latest documentation.

3 Changes

3.1 Summary of EID Issues

(EID-1341) Defaults in SamlEngine.xml

3.2 Extraction of Logging classes to the Logging module

The classes governing Logging have been extracted to the new Logging module.

3.2.1 Code changes

- Created new module EIDAS-Logging
- ConnectorIncomingEidasResponseLogger.java was moved from package eu.eidas.node.logging.connector.messages to package eu.eidas.logging.connector.messages
 - Removed now unused imports
- ConnectorIncomingLightRequestLogger.java was moved from package eu.eidas.node.logging.connector.messages to package eu.eidas.logging.connector.messages
- ConnectorOutgoingEidasRequestLogger.java was moved from package eu.eidas.node.logging.connector.messages to package eu.eidas.logging.connector.messages
- ConnectorOutgoingLightResponseLogger.java was moved from package eu.eidas.node.logging.connector.messages to package eu.eidas.logging.connector.messages
 - Removed now unused imports
- ProxyServiceIncomingEidasRequestLogger.java was moved from package eu.eidas.node.logging.service.messages to package eu.eidas.logging.service.messages
- ProxyServiceIncomingLightResponseLogger.java was moved from package eu.eidas.node.logging.service.messages to package eu.eidas.logging.service.messages
- ProxyServiceOutgoingEidasResponseLogger.java was moved from package eu.eidas.node.logging.service.messages to package eu.eidas.logging.service.messages
 - Removed now unused imports
- ProxyServiceOutgoingLightRequestLogger.java was moved from package eu.eidas.node.logging.service.messages to package eu.eidas.logging.service.messages
 - Removed now unused imports and nonfunctional code
- HashAndCounterGenerator.java was moved from package eu.eidas.node.logging.integrity to package eu.eidas.logging.integrity
- HashPatternLayoutEncoder.java was moved from package eu.eidas.node.logging.integrity to package eu.eidas.logging.integrity
- EidasRequestMessageLog.java was moved from package eu.eidas.node.logging.messages to package eu.eidas.logging.messages

- EidasResponseMessageLog.java was moved from package eu.eidas.node.logging.messages to package eu.eidas.logging.messages
- LightRequestMessageLog.java was moved from package eu.eidas.node.logging.messages to package eu.eidas.logging.messages
- LightResponseMessageLog.java was moved from package eu.eidas.node.logging.messages to package eu.eidas.logging.messages
- MessageLog.java was moved from package eu.eidas.node.logging.messages to package eu.eidas.logging.messages
- LoggingSanitizer.java was moved from package eu.eidas.node.utils to package eu.eidas.logging.utils
- A version of PropertiesUtil.java specific to properties needed for Logging was added to package eu.eidas.logging.utils
- A new BeanProvider.java was added to package eu.eidas.logging
- IFullMessageLogger.java was moved from package eu.eidas.node.logging to package eu.eidas.logging
 - Removed now unused imports
- IMessageLogger.java was moved from package eu.eidas.node.logging to package eu.eidas.logging
- LoggingConstants.java was moved from package eu.eidas.node.logging to package eu.eidas.logging
 - Removed now unused constants
- LoggingMarkerMDC.java was moved from package eu.eidas.node.logging to package eu.eidas.logging
 - Removed now unused constants
- LoggingUtil.java was moved from package eu.eidas.node.logging to package eu.eidas.logging
- MessageLoggerUtils.java was moved from package eu.eidas.node.logging to package eu.eidas.logging and modified as follows
 - Reworked the class to be passed the ProtocolEngine, CachingMetadataFetcher, FlowIdCache and CorrelationCache objects through the ApplicationContext rather than manually fetching them using the BeanProvider
 - Removed the getConnectorRedirectUrl and getProxyServiceRedirectUrl methods
 - Replaced the retrieveConnectorAttributes method with retrieveAttributes which will work for both connector and proxy
 - Removed methods getConnectorEntityId and getProxyServiceEntityId in favour of the getEntityId method
 - Reorganised the class for better readability
- All related unit tests were moved to package eu.eidas.logging
- Filter definitions and mappings were added to the web.xml for
 - ConnectorIncomingEidasResponseLoggerFilter
 - ConnectorIncomingLightRequestLoggerFilter
 - ConnectorOutgoingEidasRequestLoggerFilter
 - ConnectorOutgoingLightResponseLoggerFilter

- ProxyServiceIncomingEidasRequestLoggerFilter
- ProxyServiceIncomingLightRequestLoggerFilter
- ProxyServiceOutgoingEidasResponseLoggerFilter
- ProxyServiceOutgoingLightRequestLoggerFilter
- Modified BeanProvider.java in Connector and Proxy Service modules to ensure application context gets passed down to Logging module
- Added EIDAS-Logging as a dependency in the EIDAS-Security module

3.3 Generify Loggers and Logging Filters

In order to improve code reuse and maintainability in the logging module. The existing implementation has been generified.

3.3.1 Code changes

EIDAS-Logging

New abstract classes that implement IMessageLogger.java and IFullMessageLogger.java

- EidasRequestLogger.java
- EidasResponseLogger.java
- LightRequestLogger.java
- LightResponseLogger.java

New Filters to bind the loggers to an application

- IncomingLoggerFilter.java
- OutgoingLoggerFilter.java

Other changes

- Updated IMessageLogger loggers now log to "eu.eidas.communication.requests.**CLASS**" instead of "eu.eidas.communication.requests_**CLASS**". Similar for eu.eidas.communication.responses.
- Added MessageLoggerFilter.java as the abstract superclass for IncomingLoggerFilter and OutgoingLoggerFilter.
- Updated related unit tests.

EIDAS-Node-Connector

Loggers

- ConnectorIncomingEidasResponseLogger.java
- ConnectorIncomingLightRequestLogger.java
- ConnectorOutgoingEidasRequestLogger.java
- ConnectorOutgoingLightResponseLogger.java

Other changes

- Updated logback.xml: Additional <logger> hooks toward eIDASNodeConnectorDetail removed to prevent double logging.
- Updated web.xml: Logging filter definition replaced by IncomingLoggerFilter and OutgoingLoggerFilter with params to point to the right logger.

- Updated related unit tests.

EIDAS-Node-Proxy

Loggers

- ProxyServiceIncomingEidasRequestLogger.java
- ProxyServiceIncomingLightResponseLogger.java
- ProxyServiceOutgoingEidasResponseLogger.java
- ProxyServiceOutgoingLightRequestLogger.java

Other changes

- Updated logback.xml: Additional <logger> hooks toward eIDASNodeConnectorDetail removed to prevent double logging.
- Updated web.xml: Logging filter definition replaced by IncomingLoggerFilter and OutgoingLoggerFilter with params to point to the right logger.
- Updated related unit tests.

EIDAS-Commons

Removed IMessageLogger.java: Unused legacy class, removed to avoid confusion.

3.4 Create a contract between the app for data used in JSP from the HttpServlet Request

The values that are injected into the JSP views are now moved to a ViewMapping class per JSP. To be used in Sevlets and Loggers.

3.4.1 Code changes

EIDAS-Node-Connector

- ConnectorErrorViewMapping.java for connectorErrorPage.jsp
- ConnectorLightResponseViewMapping.java for tokenRedirectMsConnector.jsp
- ConnectorSamlRequestViewMapping.java for colleagueRequestRedirect.jsp

Updated setAttribute call in

- ColleagueResponseServlet.java
- ConnectorErrorServlet.java
- SpecificConnectorRequestServlet.java
- ConnectorOutgoingEidasRequestLogger.java
- ConnectorOutgoingLightResponseLogger.java

Other changes

- Deleted class NodeParameterNames.java
- Deleted class SamlFailureResponseException.java
- Updated Unit tests related to Loggers

EIDAS-Node-Proxy

- ProxyErrorViewMapping.java for proxyServiceErrorPage.jsp
- ProxyLightRequestViewMapping.java for tokenRedirectMsProxyService.jsp

- ProxySamlResponseViewMapping.java for connectorRedirect.jsp; samlTokenFail has been merged into samlToken

Updated setAttribute call in

- ColleagueRequestServlet.java
- ProxyServiceErrorServlet.java
- SamlResponseFailureServlet.java
- SpecificProxyServiceResponse.java
- ProxyServiceOutgoingLightRequestLogger.java
- ProxyServiceOutgoingEidasResponseLogger.java

Other Changes

- Deleted class NodeParameterNames.java
- Updated Unit tests related to Loggers

3.5 Extraction of Security classes to the Security module

The classes governing Security have been extracted to the new Security module.

3.5.1 Code changes

- Created new module EIDAS-Security
- ContentSecurityPolicyFilter was moved from package eu.eidas.node.security to package eu.eidas.security.csp
- ContentSecurityPolicyReportServlet.java was moved from package eu.eidas.node.security to package eu.eidas.security.csp
 - Removed now unused import
- RemoveHttpHeadersFilter.java was moved from package eu.eidas.node.security to package eu.eidas.security.header
- SecurityResponseHeaderHelper.java was moved from package eu.eidas.node.security to package eu.eidas.security.header
 - Removed now unused method responseHasHeaders
- AbstractNodeServlet.java was moved from package eu.eidas.node to package eu.eidas.security
 - Removed now unused methods getServletDispatcher, getApplicationContext, encodeURL, forwardRequest and invalidateEidasNodeSession
- A new BeanProvider.java was added to package eu.eidas.security
- ConfigurationSecurityBean.java was moved from package eu.eidas.node to package eu.eidas.security
 - Removed now unused method getMessageSource
- ExtendedServletResponseWrapper.java was moved from package eu.eidas.node to package eu.eidas.security
- SecurityRequestFilter.java was moved from package eu.eidas.node to package eu.eidas.security
 - Removed unused variable serialVersionUID

- SessionHolder.java was moved from package eu.eidas.node to package eu.eidas.security
- Added class SecurityError to serve as top-level exception for the Security module
- Replaced all instances of EidasNodeException with SecurityError in SecurityRequestFilter.java
- Updated unit tests to expect SecurityError instead of EidasNodeException in SecurityRequestFilterTest.java
- All related unit tests were moved to package eu.eidas.security
- Added class ContentSecurityPolicyFilterTest containing unit tests for ContentSecurityPolicyFilter
- Added class RemoveHttpHeadersFilterTest containing unit tests for RemoveHttpHeadersFilter
- Modified BeanProvider.java in Connector and Proxy Service modules to ensure application context gets passed down to Security module
- Added EIDAS-Security as a dependency in the EIDAS-Node-Connector, EIDAS-Node-Proxy, EIDAS-SpecificConnector and EIDAS-SpecificProxyService modules

3.6 Introduction of OCSP / CRL Checking

In accordance with the X.509 specifications, the eIDAS Node will now perform certificate revocation checking when the certificate in question has either a OCSP or CRL endpoint configured. Both OCSP and CRL are therefore supported by the eIDAS Node.

In order to reduce overhead, this validation will only be performed when checking the **metadata signature**. Therefore, the impact on performance will scale inversely with metadata cache expiration periods. A shorter duration will mean checking revocation lists more often, and thus a larger increase in overhead.

Revocation checking must be enabled using the configuration parameter `enable.certificate.revocation.checking`.

It is further possible to enable a 'soft fail' functionality so that certificates are assumed not to be revoked when revocation data could not be retrieved. For example, when the OCSP and the CRL endpoints cannot be reached or when the revocation data is malformed or signed with an incorrect signature. This is activated with the configuration parameter `enable.certificate.revocation.soft.fail`.

More information on this new functionality can be found in *Appendix A: Certificate Revocation Validation* of document *eIDAS-Node and SAML v2.8*

3.6.1 Code changes

- EIDAS-Encryption
 - CertificateUtil.java
 - Added new method `checkChainTrust` with a new `CertificateVerifierParams` arguments to control the behaviour of the revocation checking.
 - Removed unused method `getCertificatePair`.
 - Removed unused method `getListOfCredential`.
 - Removed unused method `getTrustedCertificate`.
 - Removed unused method `toCredential` (with `KeyInfo` parameter).

- Removed unused method `validateDigestAlgorithm`.
 - `CertificateVerifierParams.java`
 - Added new class to encapsulate configuration parameter for certificate validation with revocation checking.
 - `CertificateRevokedException.java`
 - Added new exception triggered by the `CertificateTrustVerifier` when certificate is revoked.
 - `MissingCertificateRevocationDataException.java`
 - Added new exception triggered by the `CertificateTrustVerifier` when certificate has no revocation data.
 - `UntrustedCertificateException.java`
 - Added new exception triggered by the `CertificateTrustVerifier` when certificate is not trusted (does not belongs to a Trust Anchor chain).
 - `CertificateTrustVerifier.java`
 - Interface to define an implementation of certificate trust verifier. It verifies a certificate chain belongs to a Trust Anchor and that it is not considered revoked.
 - `DSSCertificateTrustVerifier.java`
 - Added new class implementing the `CertificateTrustVerifier` interface using the DSS Library. Could throw `CertificateRevokedException`, `MissingCertificateRevocationDataException` and `UntrustedCertificateException`.
 - `NoOpRevocationDataLoadingStrategy.java`
 - Added new class used by the `DSSCertificateTrustVerifier` to disable revocation data download.
- `EIDAS-SAMLEngine`
 - `KeyStoreSignatureConfigurator.java`
 - Mapping of new property "enable.certificate.revocation.checking" inside the `SignatureConfiguration` object.
 - Mapping of new property "enable.certificate.revocation.soft.fail" inside the `SignatureConfiguration` object.
 - `SignatureConfiguration.java`
 - Added new boolean field `enableCertificateRevocationChecking` to enable/disable revocation checking on metadata certificate signature.
 - Added new boolean field `enableCertificateRevocationSoftFail` to enable/disable "soft fail" during revocation checking on metadata certificate signature.
 - `SignatureKey.java`
 - Mapping for the key "enable.certificate.revocation.checking" in the enum: `ENABLE_CERTIFICATE_REVOCATION_CHECKING`.
 - Mapping for the key "enable.certificate.revocation.soft.fail" in the enum: `ENABLE_CERTIFICATE_REVOCATION_SOFT_FAIL`.
 - `AbstractProtocolSigner.java`
 - Updated `validateMetadataSignature` to support the configuration regarding revocation checking.

- Added a new method `validateSignature` with argument `CertificateVerifierParams` to configure the behaviour of revocation checking for certificate.
 - `ProtocolSignerI.java`
 - Added a new method `validateSignature` with argument `CertificateVerifierParams` to configure the behaviour of revocation checking for certificate.
 - `DefaultSamlEngineProperties.xml`
 - Added new property `"enable.certificate.revocation.checking"` added and set to `"true"` by default.
 - Added new property `"enable.certificate.revocation.soft.fail"` added and set to `"false"` by default.
- **EIDAS-Parent**
 - Dependency `com.google.guava:guava:jar` was upgraded from version `28.1-jre` to version `32.1.3-jre`
 - Dependency `org.slf4j:slf4j-api:jar` was upgraded from version `1.7.10` to version `1.7.15`

3.6.1.1 Configuration changes

- **EIDAS-SAMLEngine**
 - `SamlEngineProperties.xml`
 - Added new property `"enable.certificate.revocation.checking"` to enable revocation checking
 - Added new property `"enable.certificate.revocation.soft.fail"` to enable soft fail for revocation checking
 - `DefaultSamlEngineproperties.xml`
 - Added property `"enable.certificate.revocation.checking"` with default value `"true"`
 - Added property `"enable.certificate.revocation.soft.fail"` with default value `"false"`

3.7 Metadata Derivatives Belong in Metadata not ProtocolProcessor

The saml engine protocol processor inside our application is initialized with a metadata Fetcher. To prevent custom methods related to metadata on the `ProtocolProcessorI` interface, a new method has been created that returns the metadata object extended with additional convenience assessors.

`HighLevelMetadataParamsI getMetadataParameters(@NonNull String metadataUrl)` throws `EIDASSAMLEngineException`;

3.7.1 Code changes

`MessageLoggerUtils.java`

- `AUCONNECTORSAML.java`: Replaced method call `getMetadataNodeCountryCode(url)` with `getMetadataParameters(issuer).getNodeCountry()`
- `AUSERVICE.java`: Removed method `getDestinationProtocolVersion(url)`
- `AbstractProtocolEngine.java`:

- Replaced method call `getEncryptionCertificate(url)` with `getMetadataParameters(requestIssuer).getEncryptionCertificate()`
- Replaced method call `getResponseSignatureCertificate(url)` with `getMetadataParameters(requestIssuer).getResponseSignatureCertificate()`
- `EidasProtocolProcessor.java`:
 - Replaced method call `getRequestSignatureCertificate(url)` with `getMetadataParameters(requestIssuer).getRequestSignatureCertificate()`
 - Replaced method call `getResponseSignatureCertificate(url)` with `getMetadataParameters(requestIssuer).getResponseSignatureCertificate()`
 - Deprecated methods `getMetadataNodeCountryCode(url)`, `getMetadataProtocolVersions(url)`, `getEncryptionCertificate(url)`, `getMetadataEncryptionHelper()`, `getMetadataSignatureHelper()`, `getRequestSignatureCertificate(url)`, `getResponseSignatureCertificate(url)`, `getSupportedAttributes(url)`
 - Removed method `isSendRequesterId(url, requesterValue)`
 - Removed instances of the `MetadataEncryptionHelper` and `MetadataSignatureHelper` class.
 - Added method `getMetadataParameters(url)`
- `EIDASSAMLEngineException.java`
 - Added Constructor with signature: `(errorCode, errorMessage, additionalInformation, cause)`
- `ProtocolEngine.java`
 - Replaced method call `getResponseSignatureCertificate(url)` with `getMetadataParameters(requestIssuer).getResponseSignatureCertificate()`
 - Replaced method call `getRequestSignatureCertificate(url)` with `getMetadataParameters(requestIssuer).getRequestSignatureCertificate()`
- `ProtocolProcessorI.java`
 - Deprecated methods `getMetadataNodeCountryCode(url)`, `getMetadataProtocolVersions(url)`, `getEncryptionCertificate(url)`, `getRequestSignatureCertificate(url)`, `getResponseSignatureCertificate(url)`
- `EidasMetadata.java`
 - Changed the type of `supportedAttributes` from `guava ImmutableSortedSet` to `java SortedSet`
- `EidasMetadataParameters.java`
 - Updated the copy constructor to take objects of interface `EidasMetadataParametersI` and added missing fields.
- `EidasMetadataRoleParametersI.java`
 - Changed the type of `getSupportedAttributes` from `guava ImmutableSortedSet` to `java SortedSet`
- Added new class `HighLevelMetadataParams.java`: Extends class `EidasMetadataParameters`
- Added new Interface `HighLevelMetadataParamsI.java`: Adds additional getter methods
- Deprecated `MetadataEncryptionHelper.java`
- Deprecated `MetadataSignatureHelper.java`

3.8 Code cleanup

3.8.1 Removal of deprecated code

Code that was deprecated in previous versions of the Eidas Node has been removed in this version.

3.8.1.1 Code Changes

- EIDAS-Commons
 - Removed classes:
 - InvalidMOAResponseException.java
 - InvalidSAMLResponseException.java
 - InvalidSessionEIDASException.java
 - LevelOfAssurance.java
 - EidasUtil.java
 - IPersonalAttributeList.java
 - PersonalAttribute.java
 - PersonalAttributeList.java
 - PersonalAttributeString.java
 - AbstractEIDASException.java
 - Removed deprecated variables userErrorCode and userErrorMessage
 - Removed deprecated constructor AbstractEIDASException(errorCode, errorMessage, cause, userErrorCodeArg, userErrorMessageArg)
 - Removed deprecated methods getUserErrorCode and getUserErrorMessage
 - AttributeUtil.java
 - Removed method checkMandatoryAttributes(final IPersonalAttributeList personalAttrList)
 - Removed method getMissingMandatoryAttributes(final IPersonalAttributeList personalAttrList)
 - EidasProtocolVersion.java
 - Removed deprecated constant PROTOCOL_VERSION_1_1
 - IAuthenticationRequest.java
 - Removed deprecated method getOriginalIssuer
 - Marked method getOriginCountryCode as deprecated
 - AbstractAuthenticationRequest.java
 - Removed originalIssuer from constructors and builders
 - Marked originCountryCode as deprecated
 - Marked method getOriginCountryCode as deprecated
 - EidasParameterKeys.java
 - Removed deprecated constant VALIDATION_BYPASS
 - Removed deprecated constant EIDAS_CONNECTOR_ACTIVE
 - Removed deprecated constant EIDAS_SERVICE_ACTIVE

- EidasStringUtil.java
 - Replaced use of Constants class with Charsets class
- IncomingRequestState.java
 - Removed method getLevelOfAssurance
 - Removed deprecated method getQaa
 - Removed method setLevelOfAssurance
 - Removed deprecated method setQaa
- RequestState.java
 - Removed deprecated method getLevelOfAssurance
 - Removed deprecated method getQaa
 - Removed deprecated method setLevelOfAssurance
 - Removed deprecated method setQaa
- AbstractAuthenticationRequestBuilderTest.java
 - Removed usage of ORIGINAL_ISSUER from tests
- AbstractAuthenticationRequestConstants.java
 - Removed constant ORIGINAL_ISSUER
 - Updated call to StringAttributeValue
- AbstractEidasAuthenticationRequestTest.java
 - Replaced originalIssuer with originCountryCode
- EidasProtocolVersionTest.java
 - Removed usage of PROTOCOL_VERSION_1_1
- DateUtilTestCase.java
 - Removed unused imports
- EIDAS-Encryption
 - CertificateUtil.java
 - Removed deprecated method createCredential(certificate, privateKey)
 - SAMLAuthnResponseEncrypter.java
 - Removed deprecated method encryptSAMLResponseAndMarshall(samlResponse, credential)
- EIDAS-Light-Commons
 - Removed class:
 - Constants.java
 - StringAttributeValue.java
 - Removed deprecated constructor StringAttributeValue(value, isNonLatinScriptAlternateVersion)
 - StringAttributeValueMarshaller
 - Replaced constructor of StringAttributeValue with the non deprecated version
 - ImmutableAttributeMapTest.java
 - Replaced calls to StringAttributeValue
 - LightRequestTest.java

- Replaced calls to StringAttributeValue
 - LightResponseTest.java
 - Replaced calls to StringAttributeValue
- EIDAS-Node-Connector
 - Removed class:
 - EidasAttributesUtil.java
 - AUCONNECTORUtil.java
 - Removed deprecated variable bypassValidation
 - Removed deprecated method loadConfigServiceUrl
 - Removed deprecated method setBypassValidation
 - Removed deprecated method isBypassValidation
 - EidasNodeMetadataGenerator.java
 - Removed deprecated method setNodeProps
 - EidasNodeValidationUtil.java
 - Removed deprecated method isRequestLoaValid
 - Removed deprecated method isLoaValid
 - PropertiesUtil.java
 - Removed deprecated method isMetadataEnabled
 - applicationContext.xml
 - Removed bypassValidation from following beans
 - springManagedAUCONNECTORUtil
 - AUCONNECTORSAMLTest.java
 - Removed deprecation on test processSpRequestBlocksOldSpecs
 - Removed all deprecated tests
 - AUCONNECTORSAMLTestCase.java
 - Replaced deprecated exception with ConnectorError
 - AUCONNECTORUtilTestCase.java
 - Removed tests for now removed methods
 - EidasNodeValidationUtilTestCase.java
 - Removed tests for now removed methods
- EIDAS-Node-Proxy
 - Removed class:
 - EidasAttributesUtil.java
 - AUSERVICE.java
 - Removed deprecated method generateSamITokenFail
 - AUSERVICESAML.java
 - Removed method generateErrorAuthenticationResponse
 - ISERVICESAMLService.java
 - Removed deprecated method generateErrorAuthenticationResponse
 - ISERVICEService.java

- Removed two deprecated methods, both generateSamlTokenFail
 - SpecificProxyServiceResponse.java
 - Removed non-functional clause using deprecated method
 - EidasNodeMetadataGenerator.java
 - Removed deprecated method setNodeProps
 - EidasNodeValidationUtil.java
 - Removed deprecated method isRequestLoaValid
 - AUSERVICESAMLTTest.java
 - Removed deprecation on test processConnectorRequestFailsFromOldSpec
 - Removed deprecated tests
 - AUSERVICETestCase.java
 - Removed tests for now removed methods
 - EidasNodeValidationUtilTestCase.java
 - Removed tests for now removed methods
- EIDAS-SAMLEngine
 - Removed classes:
 - SAMLEngineException.java
 - X500PrincipalUtil.java
 - ConfigurationKey.java
 - Removed deprecated constant EXTENSION_PROCESSOR_CONFIGURATION
 - EidasProtocolProcessor.java
 - Removed registering of deprecated namespace from method registerResponseNamespace
 - Removed deprecated methods for marshallErrorResponse, marshallRequest and marshallResponse
 - EncryptionConfiguration.java
 - Removed all deprecated constructors
 - SignatureConfiguration.java
 - Removed all deprecated constructors
 - AbstractProtocolCipher.java
 - Removed deprecated constructors
 - AbstractProtocolDecrypter.java
 - Removed deprecated constructors
 - AbstractProtocolEncrypter.java
 - Removed deprecated constructors
 - AbstractProtocolSigner.java
 - Replaced call to deprecated constructors
 - Removed method getPublicMetadataSigningCredential
 - Removed method getPublicSigningCredential

- Removed deprecated method `sign(signableObject, onlyKeyInfoNoCert)`
 - Removed method `isRequestSignWithKey`
 - Removed method `isResponseSignWith Key`
- `AbstractSamlEngineEncryption.java`
 - Removed deprecated constructors
- `SignSW.java`
 - Removed constructor `SignSW(properties)`
- `ProtocolProcessorI.java`
 - Removed deprecated methods `marshallErrorResponse`, `marshallRequest` and `marshallResponse`
- `ProtocolSignerI.java`
 - Removed deprecated methods `getPublicSigningCredential`, `sign(signableObject, onlyKeyInfoNoCert)`, `isRequestSignWithKey` and `isResponseSignWithKey`
- `AssertionUtil.java`
 - Removed deprecated method `getUniquenessIdentifier`
- `SAMLEngineUtils.java`
 - Removed deprecated methods `encode` and `getCurrentTime`
- `ProtocolEngine.java`
 - Removed method `generateResponseMessage`
 - Removed deprecated method `generateResponseErrorMessage`
- `ProtocolEngineI.java`
 - Removed deprecated method `generateResponseMessage`
 - Removed deprecated method `generateResponseErrorMessage`
- `SAMLCore.java`
 - Removed deprecated namespace `EIDAS10_RESPONSESAML_NS`
- `EidasProtocolProcessorTest.java`
 - Removed tests for now removed methods
- `AbstractProtocolCipherTest.java`
 - Replaced calls to removed constructor
- `AbstractProtocolDecrypterTest.java`
 - Replaced calls to removed constructor
- `AbstractProtocolEncrypterTest.java`
 - Replaced calls to removed constructor
- `AbstractProtocolSignerTest.java`
 - Replaced calls to removed constructor
- `ResponseTestHelper.java`
 - Replaced calls to removed method
- `ProtocolEngineFactoryTest.java`
 - Replaced calls to removed class
- `ProtocolEngineTest.java`

- Removed tests for removed methods
 - TestMDGenerator.java
 - Replaced calls to removed methods
 - EidasSAMLResponseSyntaxTest
 - Adapted test checkSaml2pHeader to check for the currently used prefix
 - SimpleBaseTest.java
 - Removed call to SAMLEngineUtils
 - SSETestUtils.java
 - Replaced calls to removed class
- EIDAS-Metadata
 - Removed class:
 - EIDASMetadataRuntimeException.java
 - CachingMetadataFetcher.java
 - Removed call to EIDASUtil
 - EIDASMetadataParameters.java
 - Removed method setAssuranceLevel
 - EIDASMetadataParametersI.java
 - Removed deprecated method setAssuranceLevel(String)
 - MetadataSignerI.java
 - Removed deprecated method getPublicMetadataSigningCredential
 - SAMLMetadataCore.java
 - Removed deprecated constant EIDAS10_RESPONSESAML_NS
 - Removed deprecated constant EIDAS10_BASE_URI
- EIDAS-SpecificCommunicationDefinition
 - LightJAXBCodecTest
 - Replaced calls to deprecated methods in StringAttributeValue

3.8.2 Deprecation of EidasNodeException

The exception class EidasNodeException was deprecated in favour of module-specific error classes

3.8.2.1 Code changes

- EIDAS-Commons
 - Deprecated EidasNodeException.java
- EIDAS-Node-Connector
 - Removed unused catch clause in EidasNodeMetadataGenerator.java
- EIDAS-Node-Proxy
 - Removed unused catch clause in EidasNodeMetadataGenerator.java
- EIDAS-SAMLEngine
 - Updated javadoc to current status in EidasProtocolProcessor.java

- Updated javadoc and removed unused throws statement in EidasAuthnRequestValidatorTest.java

3.9 Removal of legacy keystore configuration

The old method of configuring keystores in external config has been removed in favor of the new numbered notation

3.9.1 Code Changes

- EIDAS-SAMLEngine
 - KeyStoreEncryptionConfigurator.java
 - Removed method fetchLegacyConfiguration
 - KeyStoreSignatureConfigurator.java
 - Removed method fetchLegacyConfiguration
 - AbstractProtocolSignerTest.java
 - Updated method getSigningProperties to use numbered notation
 - ProtocolSignerTest.java
 - Updated method getSigningProperties to use numbered notation
 - SAMLEngineUtilsTest.java
 - Updated method getSigningProperties to use numbered notation
 - Updated method createSignedRequestThatIsMisleading to use numbered notation
 - Updated keystore configuration to use numbered notation in following files
 - EncryptModule_DOM-test_empty.xml
 - EncryptModule_DOM-test_false.xml
 - EncryptModule_DOM-test_true.xml
 - EncryptModule_Metadata.xml
 - EncryptModule_MetadataO.xml
 - SignModule_Conf1.xml
 - SignModule_Conf2.xml
 - SignModule_Conf3.xml
 - SignModule_Conf4.xml
 - SignModule_Conf5.xml
 - SignModule_Conf6.xml
 - SignModule_DOM-test.xml
 - SignModule_ECDSA.xml
 - SignModule_EidasOnly.xml
 - SignModule_Metadata.xml
 - SignModule_MetadataO.xml
 - SignModule_P11.xml
 - SignModule_Service.xml

- EIDAS-Node-Connector
 - Updated keystore configuration to use numbered notation in following test configuration files:
 - EncryptModule_Service.xml
 - SignModule_METADATA.xml
 - SignModule_METADATA_EMPTY_TRUST.xml
 - SignModule_METADATA_INTERMEDIATE_CA_ROOT_CA_TRUST.xml
 - SignModule_METADATA_INTERMEDIATE_CA_TRUST.xml
 - SignModule_METADATA_NODE_CERT_TRUST.xml
 - SignModule_METADATA_ROOT_CA_TRUST.xml
 - SignModule_METADATA_SINGLE_CERTIFICATE.xml
 - SignModule_METADATA_TC.xml
 - SignModule_METADATA_TC_WITHOUT_INTERMEDIATE_CA_CERTIFICATE.xml
 - SignModule_METADATA_TC_WITHOUT_ROOT_CA_CERTIFICATE.xml
 - SignModule_METADATA_TC_WRONG_ORDER_EXTRA_CERTIFICATE.xml
 - SignModule_METADATA_WRONG_TRUST.xml
 - SignModule_Service.xml
 - SignModule_ServiceWithAlgorithmWhitelist.xml
 - SignModule_ServiceWithMetadataAlgorithmNotInAlgorithmWhitelist.xml
 - SignModule_ServiceWithoutAlgorithmWhitelist.xml
- EIDAS-Node-Proxy
 - Updated keystore configuration to use numbered notation in following test configuration files:
 - EncryptModule_Service.xml
 - SignModule_METADATA.xml
 - SignModule_METADATA_EMPTY_TRUST.xml
 - SignModule_METADATA_INTERMEDIATE_CA_ROOT_CA_TRUST.xml
 - SignModule_METADATA_INTERMEDIATE_CA_TRUST.xml
 - SignModule_METADATA_NODE_CERT_TRUST.xml
 - SignModule_METADATA_ROOT_CA_TRUST.xml
 - SignModule_METADATA_SINGLE_CERTIFICATE.xml
 - SignModule_METADATA_TC.xml
 - SignModule_METADATA_TC_WITHOUT_INTERMEDIATE_CA_CERTIFICATE.xml
 - SignModule_METADATA_TC_WITHOUT_ROOT_CA_CERTIFICATE.xml

- SignModule_METADATA_TC_WRONG_ORDER_EXTRA_CERTIFICATE.xml
- SignModule_METADATA_WRONG_TRUST.xml
- SignModule_Service.xml
- SignModule_ServiceWithAlgorithmWhitelist.xml
- SignModule_ServiceWithMetadataAlgorithmNotInAlgorithmWhitelist.xml
- SignModule_ServiceWithoutAlgorithmWhitelist.xml

3.9.2 Configuration Changes

- EncryptModule_Connector.xml

3.10 Support of 1.4 Technical Specifications

3.10.1 Only 1 Dataset can be Requested

It is no longer possible to request both Natural and Legal person attributes in one request. Requested attributes must belong to either the Natural Person dataset OR the Legal Person dataset.

3.10.1.1 Code changes

- EIDAS-Node-Connector
 - AUCONNECTOR.java
 - Added method validateLightRequestDatasetLimit to verify either Natural person OR Legal person dataset is requested.
 - Added call to new method validateLightRequestDatasetLimit to method getAuthenticationRequest
 - AUCONNECTORTestCase.java
 - Adapted unit tests to new behaviour

3.10.2 Block Multi Dataset Request from Connectors 1.4 Spec and Higher

In accordance with the 1.4 specification, any SAML request containing both natural and legal person attributes will be blocked within the proxy module if the protocol version specified in the connector metadata is 1.4, higher, or unspecified.

3.10.2.1 Code changes

- AUSERVICISAML
 - Added validateRequestedAttributesForProtocol method to check the data sets from SAML request and protocol version from connector metadata

3.10.3 Removal of rsa-oaep-mgf1p URI

The <http://www.w3.org/2001/04/xmlenc#rsa-oaep-mgf1p> algorithm is no longer supported as per the 1.4 spec. This leaves only the <http://www.w3.org/2009/xmlenc11#rsa-oaep> algorithm for key transport.

As such, when using key transport, the externalised config for the encryption algorithms will be ignored in favour of the default, and only, algorithm. This being <http://www.w3.org/2009/xmlenc11#rsa-oaep>.

3.10.3.1 Code changes

- EIDAS-Encryption
 - SamlAuthnResponseEncrypter.java
 - Adapted method getKeyTransportEncAlgorithm to always return <http://www.w3.org/2009/xmlenc11#rsa-oaep>. Previously present code has been commented for possible reuse in a later version.

3.10.4 Block Responses with rsa-oaep-mgf1p URI coming from Proxies 1.4 Spec or higher

In compliance with the 1.4 specification, the responses coming from the proxy service module are now restricted from using the <http://www.w3.org/2001/04/xmlenc#rsa-oaep-mgf1p> algorithm for key transport or key agreement.

Consequently, any responses received from the proxy module will be blocked if they utilize this algorithm and are associated with a metadata protocol version of 1.4, higher than 1.4, or an unspecified version.

3.10.4.1 Code changes

- ProtocolEngine
 - Added checkKeyTransportAlgorithm method to check the algorithm from the response and the metadata protocol version
- EidasProtocolVersion
 - Added 1.4 protocol version

3.10.5 Update Default Protocol Version to Include 1.4

Protocol version 1.4 has been added to the default configuration for supported protocol versions in the metadata. Current supported protocol versions in 2.8 are "1.4", "1.3" and "1.2" .

3.10.5.1 Code changes

- EIDAS-Node-Connector/src/main/resources/default/eidas.xml
- EIDAS-Node-Proxy/src/main/resources/default/eidas.xml

3.11 Upgrade of Bouncy Castle library to version 1.76

Bouncy Castle has been upgraded as a dependency in the eIDAS Connector and ProxyService. If you use Bouncy Castle as a JVM Security Provider we **recommend you upgrade** to version 1.76 because of CVE-2023-33201 and CVE-2023-33202

Signed version obtained from <https://www.bouncycastle.org/download/bcprov-jdk18on-176.jar>

3.11.1 Code changes

- Changed bouncycastle.version from 1.72 to 1.76 in the pom.xml of eidas-parent

3.12 Log Details of EIDASSAMLEngineException

EIDASSAMLEngineException was reworked to take a constructor of EidasErrorKey, additional information text and an optional cause.

These values are now used when printing the stack trace, this should make it easier to track down issues reported by citizens.

Note: Because of the sporadic use of different values in previous constructors, we've not chosen to provide backwards support through deprecation old methods.

3.12.1 Code changes

Updated:

- AbstractProtocolCipher.java
- AbstractProtocolDecrypter.java
- AbstractProtocolDecrypterTest.java
- AbstractProtocolEncrypter.java
- AbstractProtocolEngine.java
- AbstractProtocolSigner.java
- AbstractProtocolSignerTest.java
- AssertionUtil.java
- AssertionUtilTest.java
- AUCONNECTORSAML.java
- AUCONNECTORSAMLTTest.java
- AUSERVICESAML.java
- AUSERVICESAMLTTest.java
- BuilderFactoryUtil.java
- CertificateValidator.java
- ConnectorErrorUtil.java
- ConnectorErrorUtilTest.java
- DefaultCoreProperties.java
- DOMConfigurationParser.java
- DOMConfigurator.java
- EidasNodeMetadataGenerator.java
- EidasNodeMetadataGeneratorTest.java
- EidasProtocolProcessor.java
- EIDASSAMLEngineException.java
- EncryptionActivationSW.java
- KeyStoreConfigurator.java
- KeyStoreContent.java
- LevelOfAssuranceRequestValidator.java
- LevelOfAssuranceResponseValidator.java

- ListKeystoreContent.java
- MessageLoggerUtils.java
- ProtocolEngine.java
- ProtocolEngineConfigurationException.java
- ProtocolEngineConfigurationFactory.java
- ProtocolEngineFactory.java
- ProtocolEngineFactoryTest.java
- ProtocolSignerTest.java
- ProxyServiceErrorUtil.java
- ProxyServiceErrorUtilTest.java
- ResponseUtil.java
- SAMLEngineUtils.java
- SAMLEngineUtilsTest.java
- SamlResponseFailureServlet.java
- XmlSchemaUtil.java

3.12.1.1 Other changes

In Eidas-Node-Proxy and Eidas-Node-Connector: Code that funneled "EidasErrorKey.errorCode" and "EidasErrors.errorCode" into "EidasErrorKey.errorCode" has been removed:

AUSERVICESAML.java

- processConnectorRequest was reworked to remove funnel code.

ConnectorErrorUtil.java

- processSAMLEngineException was reworked to remove funnel code.
- processMetadataException was split off from processSAMLEngineException

ProxyServiceErrorUtil.java

- processSAMLEngineException was reworked to remove funnel code.
- processMetadataException was split off from processSAMLEngineException

3.13 Sign with provider that loads the keys

When a keypair is configured for signing, the application will first try to sign with the algorithms from the provider responsible for loading the keystore containing this keypair.

This is useful with allround providers such as HSM but not useful with task specific providers such as Sun, SunEC, SunRsaSign.

When the try fails, the order of security providers will be followed similar to 2.7.0

3.13.1 Code changes

eIDAS Saml Engine

Updated to accommodate passing the **ProtocolSignerI** parameter: DOMConfigurator.java, EidasExtensionConfiguration.java, EidasProtocolProcessor.java, ProtocolProcessorI.java, ProtocolEngineFactory.java

Create the new class **KeyContainerEntry**.java to bundle a PrivateKeyEntry and keyProvider together in KeyContainer.java, KeyStoreConfigurator.java, KeyStoreContent.java,

ListKeystoreContent.java

Added fields signatureKeyProvider and metadataSignatureKeyProvider to **SignatureConfiguration**.java used in KeyStoreSignatureConfigurator.java

Added **getSecurityProvider** method to SignatureMarshaller.java to check if the algorithm required exists on the security provider.

Added getSecurityProvider method to **SignatureMarshaller**.java to check if the algorithm required exists on the security provider.

Updated the createXMLSignature to the getSecurityProvider.

Updated tests.

3.14 Other changes requiring no action

3.14.1 Defaults in SamlEngine.xml

The inclusion of "fileConfiguration" in SamlEngine.xml has been made optional since most engine fields have default values.

3.14.1.1 Code changes

Removed Empty check in DOMConfigurator.java

Added testing in SamlEngine.xml and SimpleBaseTest.java

3.14.2 Upgrade dependency of xmlsec for CVE-2023-44483

The version of the xmlsec dependency was upgraded from version 2.2.3 to version 2.3.4

3.14.2.1 Code changes

upgraded the version of the xmlsec dependency in the pom.xml of the Eidas-Parent module

3.14.3 Reduce the declaration of namespaces in metadata XML

Declaration of the urn:oasis:names:tc:SAML:2.0:assertion namespace for supported attributes was moved to their parent, the IDPSSODescriptor, to prevent repetition.

3.14.3.1 Code changes

Declaring urn:oasis:names:tc:SAML:2.0:assertion on the IDPSSODescriptor in EidasMetadata and updating tests in EidasNodeMetadataGeneratorTest.java.

3.14.4 Reduce the declaration of `http://www.w3.org/2001/XMLSchema-instance` in Response Attributes

Declaration of the xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" namespace in SAML Response was moved to the document root, to prevent repetition.

3.14.4.1 Code changes

Declaration of the namespace was moved to the root of the document, and related test EidasSAMLResponseSyntaxTest was updated to this new behaviour

3.14.5 SignModule_Connector_EC.xml and SignModule_Service_EC.xml config files do not contain metadata.signature.algorithm property

Updated example configuration for metadata signing algorithm with ECDSA.

3.14.5.1 Code changes

Updated SignModule_Connector_EC.xml

Updated SignModule_Service_EC.xml

3.14.6 Replace SignatureConfiguration constructor by Builder

All usages of the SignatureConfiguration constructor outside of the SignatureConfiguration class have been replaced by calls to the SignatureConfiguration builder.

3.14.6.1 Code changes

- SignatureConfiguration
 - Added builder constructor which accepts a SignatureConfiguration argument
 - Removed nullable annotation from all setter methods
- AbstractProtocolSigner
 - Replaced complex call to constructor with call to builder with SignatureConfiguration argument
 - Deprecated unused constructors

3.14.7 Binary light token hash algorithm and provider are now a property of LightRequest and LightResponse Loggers

For calculating the Binary Light Token (BLT) hash, the code no longer uses hardcoded configuration keys inside LoggingUtil.java, instead the values can be set on the loggers. In the eIDAS Connector and Proxy this is done inside the application context with `logging.hash.digest.algorithm` and `logging.hash.digest.provider`.

3.14.7.1 Code changes

Updated method "hash" in EidasDigestUtil to omit provider when provider is null or empty

Added properties loggingHashDigestAlgorithm and loggingHashDigestProvider to

LightRequestLogger.java and LightResponseLogger.java

Added parameters digestAlgorithm and digestProvider to method createBltHash replacing use of PropertiesUtil.

Removed propertiesUtil.

Updated Connector applicationContext.xml to use setting "logging.hash.digest.algorithm" for loggingHashDigestAlgorithm and logging.hash.digest.provider for loggingHashDigestProvider

Updated Connector default/eidas.xml setting default value for logging.hash.digest.provider to empty

Updated ProxyService applicationContext.xml to use setting "logging.hash.digest.algorithm" for loggingHashDigestAlgorithm and logging.hash.digest.provider for loggingHashDigestProvider

Updated ProxyService default/eidas.xml setting default value for logging.hash.digest.provider to empty

3.14.8 Check Representative for MDS

Added a validation to check if the representative dataset contains all mandatory elements of the MDS.

3.14.8.1 Code changes

- EIDAS-SAMLEngine
 - EidasProtocolProcessor.java
 - Adapted method checkMandatoryAttributes to include checking for mandatory representative dataset, and to verify both representative and natural/legal MDS are present in response.
- EIDAS-Node-Proxy

- AUSERVICE.java
 - Corrected thrown error when failing on checking mandatory attribute set to be consistent with other instances

3.14.9 Update the Application Identifier to 2.8.0

The default application identifiers have been updated to version 2.8.0

3.14.9.1 Configuration changes

- EIDAS-Node-Connector
 - resources/default/eidas.xml
 - Application identifier was updated to 2.8.0
- EIDAS-Node-Proxy
 - resources/default/eidas.xml
 - Application identifier was updated to 2.8.0

3.14.10 Set Logback.xml verbosity to Production

Replaced DEBUG root level with INFO to be more suitable for production environment in logback.xml from connector and proxy modules.

3.14.11 Create tests for missing entries in Error properties

Tests have been added based on EidasErrorKey enum to check for missing entries in EidasErrors.properties and sysadmin.properties Added missing key invalid.light.token as Communication Error 000012 Added missing key audience.restriction.error as eIDAS SAML error 203028

3.14.11.1 Code Changes

Updated EidasErrorKey.java: Removed boilerplate javadoc and added 4 new keys

Updated eidasErrors.properties: Removed validation properties (wrong file) and added 4 new keys

Updated EidasErrorsTest.java: Added forward and backward check toward EidasErrorKey

eidas-node-connector

Updated ConnectorErrorUtilTest.java: Replaced property mocking by real properties, added test for all keys with processSAMLException

Added messageSource-test-context.xml

Updated sysadmin.properties: Added 3 missing keys

eidas-node-proxy

Added messageSource-test-context.xml

ProxyServiceErrorUtilTest.java : Replaced property mocking by real properties, added test for all keys with processSAMLException

Updated sysadmin.properties: Added 2 missing keys

eidas-security

Updated SecurityRequestFilter.java: removed dynamic keying

3.14.12 Example HSM P12 example missing metadata.signature.algorithm for RSA

Example configurations for metadata.signature.algorithm were added in the HSM example configuration files.

3.14.12.1 Configuration changes

- Added example configuration for metadata.signature.algorithm in SignModule_Connector_HSM_P12.xml

- Added example configuration for metadata.signature.algorithm in SignModule_Service_HSM_P12.xml

3.14.13 Remove password from configuration file

Removed username and password in plain text from tomcat maven plugin from parent pom.xml due to pen test report.