eIDAS-Node Migration Guide v2.6 (pre-release)

eIDAS eID Implementation

Exported on 11/17/2021

Table of Contents

Metadata signing certificate chain should not be extrapolated from keystore contentDocument history

Version	Date	Modification reason	Modified by
1.0	17/11/2021	Information how to migrate from eIDAS-Node v2.5 to eIDAS-Node v2.6	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.

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 v2.5 to eIDAS-Node v2.6.

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 2.6.
Chapter 3 — *Changes*: Contains detailed information about the changes that should be taken into consideration when migrating to eIDAS-Node version 2.6.

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 2.6, 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 License (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 2.6 you should have:

- already implemented eIDAS-Node version 2.5;
- downloaded the eIDAS-Node v2.5 Integration Package; and
- downloaded the latest documentation.

3 Changes

3.1 Summary of changes

(EID-1086) The EidasNode 2.4. version breaks metadata signing using HSM

(EID-1090) Decryption uses all keys in keystore to decrypt

(EID-1126) IEIDASLogger could be removed

(EID-1129) Improve the documentation of installation and configuration guide

(EID-1163) AssertionUtil issue with the enableAddressAttributeSubjectConfirmationData field

(EID-1188) Validation of representation response only allows two types of persons

(EID-1189) Deployment of standalone eIDAS node should not require specific connector and proxy configuration

(EID-1191) Deployment of the Specific Proxy should not require specific connector and eidas configuration startup parameters

3.2 Implementation of support for PKCS11 AKA "HSM"

The support to use PKCS11 was added to the CEF eID Node's code. This implementation also addresses issue EID-1086.

3.2.1 Code changes

It is now possible to use PKCS11 keystores, the KeyStoreConfigurator class was updated both

- to allow empty keyStorePath configuration entry when together with keyStoreType equals to PKCS11
- and to correctly load PKCS11 keystore

As part of the implementation of the support for the signing using from PKCS11, some changes were done:

- Creation of SignatureMarshaller class to override the SignatureMarshaller from Opensaml to allow an adaptation of the JCEMapper registered algorithm when signing with PKCS11.
- Tests added to in SignatureMarshallerTest.
- Modification of the EidasExtensionConfiguration class, with changes in the "*configureExtenstion(ProtocolProcessorI protocolProcessor)*" method to register the SignatureMarshaller as the ObjectProvider in the XMLObjectProviderRegistrySupport in the process and with the creation of new methods ("*overrideJCEMapperForPKCS11()*" and "*resetJCEMapper()*") to make the adaption of the JCEMapper algorithm registry. Tests added to in EidasExtensionConfigurationTest.
- The AbstractProtocolSigner class was also modified to be able to get the digest algorithm correctly even without the BouncyCastle provider installed.
- In this context, two pom.xml were also modified to remove the "org.opensaml:opensaml-core" unneeded dependency in the following modules :
  - EIDAS-Node-NoJcache
  - EIDAS-Node-Web-Jar
- Additionaly tests added in KeyStoreType and KeyStoreConfigurator.

As part of the implementation of the support for the encryption using keys from PKCS11, we decided to remove the usage of the "se.swedenconnect.opensaml:opensaml-security-ext" library. This removal is due to the addition of classes supporting the key agreement directly in the Opensaml libraries (opensaml-xmlsec-api, opensaml-xmlsec-impl, opensaml-security-impl, ...). Therefore, the version of the opensaml libraries used is 4.1.1.

In the eIDAS-Node code, we needed to adapt both our SAMLAuthnResponseEncrypter and SAMLAuthnResponseDecrypter to replace the usage of the opensaml-security-ext by the classes from the opensaml-xmlsec-impl dependency.

Some new classes were also added :

- the EidasKeyInfoCredentialResolver class to map local credential with RecipientKeyInfo in Elliptic Curve encryption scenario. Tests added to EidasKeyInfoCredentialResolverTest.
- the EvaluableX509CertificatesCredentialCriterion class to be able to check if two credential are the same based on there x509 certificate. Tests added to EvaluableX509CertificatesCredentialCriterionTest.
- the EidasECDHKeyAgreementProcessor class which is an extension of the ECDHKeyAgreementProcessor from opensaml which will also handle PKCS11 credential. Tests added to EidasECDHKeyAgreementProcessorTest.
- the Pkcs11Decrypter which enable a key decryption process when using PKCS11 keystores. Tests added to Pkcs11DecrypterTest. Tests added to Pkcs11DecrypterTest.
- the FirstInlineEncryptedKeyResolverTest was added to cover the code present in the FirstInlineEncryptedKeyResolver class.

The DecryptionUtils class had also several changes mainly due to the replacement of library usage (from opensaml-security-ext to opensaml-xmlsec-impl).
The method "public static EidasDecryptionConfiguration buildDefaultDecryptionConfiguration(List<Credential> credentials)" was modified not requiring any credentials anymore as well as the method "static List<KeyInfoProvider> buildDefaultKeyInfoProviders(List<Credential> credentials)" which not requiring any credentials anymore.

Additional new testclass helpers were added:

- DecrypterHelper.java
- EncrypterHelper.java

Please note that the decryption using RSA with PKCS11 cannot be performed, since the SunPKCS11 provider does not provide necessary mechanism CKM_RSA_PKCS_OAEP as it is not present in the list at https://docs.oracle.com/en/java/javase/11/security/pkcs11-reference-guide1.html#GUID-D3EF9023-7DDC-435D-9186-D2FD05674777

3.2.2 Configuration changes

Of course, to use the keys from a PKCS11 keystore, some configuration changes are needed in e.g. files

- SignModule_Connector_EC.xml or SignModule_Connector.xml
- SignModule_Service_EC.xml or SignModule_Service.xml
- EncyrptModule_Connector.xml
- EncyrptModule_Service.xml

for example a possible change would be

```
<!-- Request signature -->
<entry key="keyStorePath">NONE</entry>
<entry key="keyStorePassword">0001password</entry>
<entry key="keyPassword">0001password</entry>
<entry key="issuer">CN=local-demo-cert, OU=DIGIT, O=European Commission,
L=Brussels, ST=Belgium, C=BE</entry>
<entry key="serialNumber">6C4536E0B25A05E8214A46C9086521977045417E</entry>
<entry key="keyStoreType">PKCS11</entry>
<entry key="keyStoreProvider">SunPKCS11-ykcs11</entry>
```

Please note that, the details expressed here, depend on the actual PKCS11 implementation used. Furthermore, in the released demo configuration files, we still use PKCS12 keystores so this type changes will not be appearing.

3.3 Modification of BouncyCastle's security provider installation through code

Due to the addition of the support for HSMs, since other security providers might be needed, it was decided to remove the in-code enforcement of the usage of the BouncyCastle provider.

3.3.1 Code changes

The *BLOCK_BOUNCY_CASTLE_PROVIDER_REINSTALL("block.security.provider.reinstall")* property was removed from the *EidasParameterKeys* class since the property is not needed anymore. The *EidasDefaultSecurityConfiguration* class was also modified to remove all code concerning the installation or update of the BouncyCastle provider. The *AbstractProtocolSigner* and the *AbstractProtocolCipher* classes have been modified to remove the static initialization block which was enforcing the installation of the BouncyCastle provider as security provider. The *BouncyCastleBootstrap class* was removed. Some tests classes was modified to install the BouncyCastle provider before execution of the tests and to remove the BouncyCastle provider after execution, here is the list of modified test classes :

- EIDAS-Encryption/src/test/java/eu/eidas/encryption/*SAMLAuthnResponseEncrypterTestConfig.java*
- EIDAS-Node/src/test/java/eu/eidas/node/auth/metadata/*EidasNodeMetadataGeneratorTest.java*
- EIDAS-Node/src/test/java/eu/eidas/node/auth/metadata/*TestEidasNodeFileMetadataProcessor.java*
- EIDAS-Node/src/test/java/eu/eidas/node/auth/metadata/*TestEidasNodeFileMetadataProcessorTrustChain.java*
- EIDAS-Node/src/test/java/eu/eidas/node/auth/metadata/*TestEidasNodeMetadataLoader.java*
- EIDAS-Node/src/test/java/eu/eidas/node/auth/metadata/*TestEidasNodeMetadataTrustChain.java*
- EIDAS-SamlEngine/src/test/java/eu/eidas/auth/engine/core/impl/*AbstractProtocolDecrypterTest.java*
- EIDAS-SamlEngine/src/test/java/eu/eidas/auth/engine/core/impl/*AbstractProtocolSignerTest.java*
- EIDAS-SamlEngine/src/test/java/eu/eidas/auth/engine/xml/opensaml/*SAMLEngineUtilsTest.java*
- EIDAS-SamlEngine/src/test/java/eu/eidas/auth/engine/xml/opensaml/*XmlSchemaUtilTest.java*
- EIDAS-SamlEngine/src/test/java/eu/eidas/auth/engine/*AbstractProtocolEngineTest.java*

And the *BouncyCastleBootstrapTest class* was removed since the *BouncyCastleBootstrap* class was also removed.

3.3.2 Configuration changes

Since the code is not configuring the security provider, the configuration of the BouncyCastleProvider needs to be added to the *java.security* file.

For more information, see https://docs.oracle.com/en/java/javase/11/security/howtoimplaprovider.html#GUID-FB9C6DB2-DE9A-4EFE-89B4-C2C168C5982D

3.4 Upgrade from OpenSaml 3 to OpenSaml 4

The OpenSaml dependencies were upgrade from version 3.4.3 to version 4.1.1. This implementation involved several code and configuration changes which are described in the following sub-sections.

3.4.1 Code Changes

In the context of the migration to OpenSaml 4, multiple dependencies needed to be updated :

1. The *net.shibboleth.utilities:java-support* dependency
    a. Version was changed to version 8.0.0 to be compatible with OpenSaml 4.
    b. It was also necessary to add the repository definition in the EIDAS-SamlEngine/pom.xml and in the EIDAS-Node/pom.xml
2. The *org.opensaml* dependencies
    a. The version of the opensaml dependencies was changed from version 3.4.3 to version 4.1.1.
3. The *se.swedenconnect.opensaml:opensaml-security-ext*
    a. Changed from version 1.0.7 to version 2.0.1 to be compatible with OpenSaml 4
    b. And then removed since key agreement support was added in OpenSaml 4.1.1 (see section Implementation of support for encrypting with HSM keys)
4. The *org.opensaml:opensaml-security-api* and the *org.opensaml:opensaml-security-impl* dependencies
    a. Were added to
        i. the EIDAS-Parent/pom.xml
        ii. and the EIDAS-Encryption/pom.xml
5. The *org.eclipse.persistence:org.eclipse.persistence.moxy* dependency
    a. Version was changed from version 2.6.3 to version 2.7.8 in the SimpleProtocol pom.xml
    b. Version of the SimpleProtocol module was therefore also changed to version 0.0.4
6. The *org.mockito:mockito-core* dependency
    a. Version was changed from version 2.13.0 to version 3.7.0
7. The *org.codehaus.mojo:jaxb2-maven-plugin* dependency
    a. Version was changed from version 2.3.1 to version 2.5.0
8. The *org.apache.maven.plugins:maven-compiler-plugin* dependency
    a. Version was changed from version 2.3.2 to version 3.8.0
    b. Configuration was changed from Java 1.8 to Java 11

Due to this dependencies changes, the code had to be modified to fix some compiling issues :

1. Removal of AbstractSAMLObject from OpenSaml, due to this removal all the classes that were extending that class in the Node code had to be modified to extends the AbstractXMLObject class instead. Here is the list of modified classes :
    a. EIDAS-Metadata/src/main/java/eu/eidas/auth/engine/metadata/samlobjects/NodeCountryTypeImpl.java
    b. EIDAS-Metadata/src/main/java/eu/eidas/auth/engine/metadata/samlobjects/SPTypeImpl.java
    c. EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/core/eidas/impl/DigestMethodImpl.java
    d. EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/core/eidas/impl/RequestedAttributeImpl.java
    e. EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/core/eidas/impl/RequestedAttributesImpl.java
    f. EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/core/eidas/impl/SigningMethodImpl.java

2. Replacement of the usage of Joda time library by java time library by OpenSaml, therefore some methods signatures in OpenSaml had changed. To resolve this, a conversion from joda time date information is applied. For this reason, the following classes have been modified :
    a. EIDAS-Metadata/src/main/java/eu/eidas/auth/engine/metadata/EidasMetadata.java
    b. EIDAS-Metadata/src/main/java/eu/eidas/auth/engine/metadata/MetadataUtil.java
    c. EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/xml/opensaml/AssertionUtil.java
    d. EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/xml/opensaml/BuilderFactoryUtil.java
    e. EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/xml/opensaml/ResponseUtil.java
    f. EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/core/eidas/EidasProtocolProcessor.java
    g. EIDAS-SAMLEngine/src/test/java/eu/eidas/auth/engine/core/validator/eidas/EidasAuthnRequestValidatorTest.java
    h. EIDAS-SAMLEngine/src/test/java/eu/eidas/auth/engine/core/validator/eidas/EidasResponseValidatorTest.java
    i. EIDAS-SAMLEngine/src/test/java/eu/eidas/auth/engine/xml/opensaml/ResponseTestHelper.java

Additionally, due to the removal of the *org.apache.http.conn.ssl.X509HostnameVerifier*, the BaseMetadataFecther class was also modified and this was modified by the use of javax.net[1]*.ssl.HostnameVerifier*

it was necessary to add JSON dependencies due to upgrade to Java 11 requirement. Therefore the dependency org.glassfish:javax.json was added to the following :

- EIDAS-Parent/pom.xml
- EIDAS-SpecificConnector/pom.xml
- EIDAS-SpecificProxyService/pom.xml
- EIDAS-IdP-1.0/pom.xml
- EIDAS-SP/pom.xml

Likewise, it was also necessary to add jaxb dependencies due to the removal of the support for JAXB in java 11. Therefore the *jakarta.xml.bind:jakarta.xml.bind-api:2.3.3* and *org.glassfish.jaxb:jaxb-runtime:2.3.3* dependencies were added to the following poms :

- EIDAS-Parent/pom.xml
- EIDAS-Light-Commons/pom.xml
- EIDAS-SpecificCommunicationDefinition/pom.xml
- EIDAS-SpecificConnector/pom.xml

In order to fix the unit tests when using the java 11, the following classes were modified :

- *XmlExternalEntityTest* : Adapted tests "testTransformShowXXE"
- *CountryCodesTest* : Adapted list of country codes to remove the Netherlands Antilles
- *EidasCipherSuiteTest* : Adapted the test method "JdkSupport" and removed references to JDK 7 and JDK 8.
- *JdkVersion* : Adapted the enum to have only new constant for the JDK 11 and removed the ones for JDK 7 and 8.

Seeing as the migration to OpenSaml 4 requires an upgrade to Java 11, some few changes were done so the Ignite configuration can work. In order for the code to work with Ignite when running on wildfly server with Java 11, the following dependency had to be added in the "jboss-deployment-structure.xml".

```
<dependencies>
    <module name="jdk.unsupported" />
</dependencies>
```

3.4.2 Configuration changes

Seeing as the migration to OpenSaml 4 requires an upgrade to Java 11, the supported list of servers has been changed with server that can support java 11.

The supported list of servers is now the following :

- Tomcat v9.0.x
- Wildfly 23.0.2 Final (Servlet Distribution)
- Weblogic 14.1.1.0.0
- WebSphere Liberty 21.0.0.5 (WebProfile 8)

For wildlfy, the AdditionalFiles folder the folders Wildfly11 and Wildfly15 were removed and one for Wildfly23 was added.

Note that glassfish is not supported in this version so the glassfish maven profiles and web configuration files have therefore been removed from the project.

[1] http://javax.net

3.5 Removal of Hazelcast support

Hazelcast and the JCache provider module for Hazelcast have been removed because of the end of support lifecycle of the Hazelcast 3.x version and the incompatibility with current dependencies in the node. Furthermore, for the newer versions of Hazelcast, an enterprise license is required to be allowed to use key features such as TLS. Since eIDAS Node version 2.3.0 we provide an example how to use Apache Ignite as a distributed cache provider.

3.5.1 Code changes

Therefore the following modules have been removed from the project :

- EIDAS-JCache-Hazelcast
- EIDAS-JCache-Hazelcast-Node
- EIDAS-JCache-Hazelcast-Specific-Communication

The maven profiles, that allowed to build the node with the Hazelcast modules, have also been removed as well as the demo configuration files from EIDAS-Config/server (*hazelcastNode.xml* and *hazelcastSpecificCommunication.xml*). Some configuration parameters, specific to hazelcast, have also been removed from some test configuration files (*specificCommunicationDefinitionConnector.xml* and *specificCommunicationDefinitionProxyService.xml* in EIDAS-SpecificCommunication module).

3.6 eIDAS Node default parameters

To simplify the eIDAS Node default parameters configuration several entries were moved from external configuration files into internal configuration files inside the Eidas-Node module.

3.6.1 Code changes

EIDASValues class was also modified with the removal of some entries which are considered only as parameter keys and not values, here is the list of the removed constant variables :

- METADATA_ACTIVE
- EIDAS_METADATA_HIDE_LOATYPE
- DISABLE_CHECK_MANDATORY_EIDAS_ATTRIBUTES
- VALIDATION_SUFFIX

EidasNodeMetadataGenerator class was modified to remove the code supporting an alternative way of publishing the LoA attribute values in the metadata. It will not be able anymore to publish them with an xsi:type="xs:string". AUCONNECTORUtil class has been modified with deprecation of methods related to the *bypassValidation* attribute since it isn't used by the code. Constants with value "hashDigest.className" were removed from both EidasParameterKeys and EIDASValues and were replaced by constants for "logging.hash.digest.algorithm" and for "logging.hash.digest.provider" were added in EidasParameterKeys. Implementation in LoggingUtil was changed to use EidasDigestUtil#hash method instead of EidasDigestUtil#hashPersonalToken. Note that for the case of "logging.hash.digest.provider" no value was added to default eidas.xml, the application default provider will be used but a specific provider can be added in the external eidas.xml. The deprecated method hashPersonalToken(byte[] samlToken, String className) was removed. AUCONNECTOR and AUSERVICECitizen were modified to remove the support of the "disable.check.mandatory.eidas.attributes" config parameter. Unit tests in AUCONNECTORTestCase were therefore also adapted. AUCONNECTORSAMLTestCase was also modified due to the removal of the VALIDATION_SUFFIX constant. In the class EidasParametersKeys, the constant EIDAS_BYPASS was renamed to VALIDATION_BYPASS. FileMetadataLoader was updated to not load static metatdata when the metadata.file.repository is not defined or empty in eidas.xml

3.6.2 Configuration changes

The following entries were removed from external eidas.xml and moved into /EIDAS-Node/src/main/resources/default/eidas.xml:

- tls.enabled.ciphers
- tls.enabled.protocols
- security.header.CSP.enabled
- security.header.CSP.includeMozillaDirectives
- security.header.XXssProtection.block
- security.header.XContentTypeOptions.noSniff
- security.header.XFrameOptions.sameOrigin
- security.header.HSTS.includeSubDomains
- trusted.sp.domains
- validation.bypass
- validation.method
- max.requests.sp
- max.time.sp
- max.requests.ip
- max.time.ip
- eidas.protocol.version
- eidas.application.identifier
- saml.connector
- saml.service
- active.module.connector
- active.module.service
- check.citizenCertificate.serviceCertificate
- insert.prefix.identifiers.country.code
- validate.prefix.country.code.identifiers
- metadata.activate
- metadata.http.retrieval
- metadata.validity.duration (Value changed to 86400 to align with the documentation)
- metadata.restrict.http
- metadata.check.signature
- nonDistributedMetadata.retention
- saml.audit
- full.message.logging
- allow.redirect.binding
- validate.binding
- service.LoA

Only if the values need to be overridden should the operators of the eIDAS Node set this in the external eidas.xml.

The following entries were just removed from external eidas.xml without being copied to the default configuration :

- metadata.file.repository
- metadata.hide.loatype (this property is not supported anymore)
- metadata.sector
- node.metadata.not.signed.descriptors
- disable.check.mandatory.eidas.attributes (this property is not supported anymore)
- DEMO-SP.validation

- DEMO-SP-CB.validation
- DEMO-SP-CC.validation
- DEMO-SP-CD.validation
- DEMO-SP-CE.validation
- DEMO-SP-CF.validation

In some cases, when default values were associated to the configuration entries in the applicationContext.xml, since it can now be avoided, the applicationContext.xml was modified with the removal of those inline default value.

The support of the following configuration entry was removed :

- hashDigest.className

New configuration entries were added :

- logging.hash.digest.algorithm (in replacement of hashDigest.className)
- logging.hash.digest.provider (in replacement of hashDigest.className) no value was added to default eidas.xml, but a specific provider can be added in the external eidas.xml using this property.

3.7 Remove stork's QAA related code

In order to prepare for the removal of the old QAA codes (replaced by levels of assurance), several methods and fields have been deprecated to be removed in a later version.

3.7.1 Code changes

In interface RequestState and implementing class IncomingRequestState, following methods were deprecated:

- getQaa
- setQaa

Furthermore, in IncomingRequestState, the qaa fields were removed from all builders

QAA fields and related methods were removed in the following classes:

- EidasErrorKey
- EidasParameterKeys
- NodeParameterNames

Following methods were changed to accommodate this removal

- processSpRequest in AUCONNECTORSAML
- validateSP in AUCONNECTORUtil was changed and renamed to validateRequestHasLoas
- isValidQAALevel, isValidSPQAALevel, getMinQAA, setMinQAA, setMaxQAA, getMaxQAA, in AUCONNECTORUtil
- `getMaxQAAlevel, setMaxQAAlevel, getMinQAA, setMinQAA, getMaxQAA, setMaxQAA, in AUSERVICESAML and associated fields`

Additionally QAA related error keys were removed from the following files:

- eidasErrors.properties
- eidastranslation.properties
- errors.properties
- sysadmin.properties
- package.properties
- package_en.properties

Related tests were also updated in following classes:

- AUCONNECTORSAMLTestCase
- AUCONNECTORUtilTestCase
- AUSERVICESAMLTestCertif
- AuthRequestSignatureTest
- AuthResponseTest
- AuthenticationResponseTest
- EidasAuthRequestSignatureTest
- EidasAuthRequestTest
- EidasAuthResponseTest
- EidasMessageFormatOnlyTest
- SAMLEngineTimeSkewTest
- AUSERVICETestCase
- TestingConstants

Adapted applicationContext.xml to changes in the described classes.

3.7.2 Configuration Changes

the following entries were removed from eidas.xml

- service.maxQAAlevel
- min.qaaLevel.value
- max.qaaLevel.value

3.8 Improvement of (default) configuration for SAML engine

An improvement was done to make the SAML engine module have default configuration functionality. This allows to reduce the need for explicit configuration in both SAML Engine and Node external configuration files. However, it removes the possibility to configure the SAML engine using node external configuration files, such as eidas.xml. Note that this implementation also addresses issue EID-1163.

3.8.1 Code changes

The following code changes were done:

1. applicationContext.xml
   a. removed the value from the second constructor-arg of NodeProtocolEngineConfigurationFactory bean
2. default/defaultSamlEngineProperties.xml
   a. new file added to allow setting of internal default properties for the SAML Engine Module
3. DOMConfigurator.java
   a. added method getDefaultSamlProperties, and calling of that method inside createReloadableProxyIfNeeded method
4. ExternalConfigurationFileAccessor.java
   a. renamed overrideParameters to defaultParameters and change order of setting to be the one expected for default parameters
5. ProtocolEngineFactoryTest.java
   a. updated the tests to new default configuration
6. SamlEngine_DOM-test_false.xml
   a. new file added to be used in ProtocolEngineFactoryTest.java

7. ProtocolEngineI
    a. deprecated method generateResponseMessage(@Nonnull IAuthenticationRequest request, @Nonnull IAuthenticationResponse response, boolean signAssertion, @Nonnull String ipAddress)
    b. added new method generateResponseMessage(@Nonnull IAuthenticationRequest request, @Nonnull IAuthenticationResponse response, @Nonnull String ipAddress)
8. AUSERVICESAML.java
    a. removed getting of response.sign.assertions value
9. AssertionUtil
    a. Removed unconfigurable code that was supposed to limit the addition of the ipAddress to the subject confirmation data
10. AssertionUtilTest
    a. Removed irrelevant tests (*setEnableAddressAttributeSubjectConfirmationDataEnabled(...) and setEnableAddressAttributeSubjectConfirmationDataDisabled(...))*
11. EidasProtocolProcessor
    a. Added if condition in the method *marshallResponse(@Nonnull IAuthenticationRequest request, @Nonnull IAuthenticationResponse response, @Nonnull String ipAddress, @Nonnull SamlEngineCoreProperties coreProperties, @Nonnull DateTime* currentTime) to whether or not provide the ipAddress to generate the assertions.
12. EidasProtocolProcessorTest
    a. Added tests for the method *marshallResponse(@Nonnull IAuthenticationRequest request, @Nonnull IAuthenticationResponse response, @Nonnull String ipAddress, @Nonnull SamlEngineCoreProperties coreProperties, @Nonnull DateTime* currentTime)
13. SamlEngineCoreProperties
    a. Added in the interface a constant "ENABLE_ADDRESS_ATTRIBUTE_SUBJECT_CONFIRMATION_DATA" for the configuration key associated.
14. EidasNodeMetadataGenerator
    a. Updated the method *generateMetadata* to get the signature algorithm from the protocol engine instead of from the node properties.
    b. Updated the method generateMetadata to get the encryption algorithms whitelist for the protocol engine instead of from the node properties
15. EidasNodeMetadataGeneratorTest
    a. Updated the method *testGenerateMetadataWithUnsupportedSigningAlgorithms*
16. KeyStoreSignatureConfigurator
    a. Updated the method *getSignatureConfiguration* to adapt/filter signature algorithm whitelist to authorized algorithms only.
17. ProtocolSignerI
    a. Added method *getSignatureConfiguration()*
18. AbstractProtocolSigner
    a. Updated implementation of the class to be able to implement the new method *getSignatureConfiguration()* of the ProtocolSignerI
19. AbstractProtocolSignerTest
    a. Updated tests dealing with empty or null value of the signature algorithm since it is not allowed anymore to have this.
20. SignatureConfiguration
    a. Add a method *getSignatureAlgorithmWhitelist()* that return the signature algorithm whitelist as an *ImmutableSet*
21. TestMDGenerator
    a. Remove constant TEST_SIGNATURE_ALGORITHM from class
22. Multiple tests configuration files were modified
    - EIDAS-Node/src/test/resources/EncryptModule_Service.xml
    - EIDAS-Node/src/test/resources/SignModule_METADATA.xml
    - EIDAS-Node/src/test/resources/SignModule_METADATA_EMPTY_TRUST.xml
    - EIDAS-Node/src/test/resources/SignModule_METADATA_INTERMEDIATE_CA_ROOT_CA_TRUST.xml

- EIDAS-Node/src/test/resources/SignModule_METADATA_INTERMEDIATE_CA_TRUST.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_NODE_CERT_TRUST.xml
- EIDAS-Node/src/test/resources/SignModule_ROOT_CA_TRUST.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_SINGLE_CERTIFICATE.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_TC.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_TC_WITHOUT_INTERMEDIATE_CA_CERTIFICATE.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_TC_WITHOUT_ROOT_CA_CERTIFICATE.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_TC_WRONG_ORDER_EXTRA_CERTIFICATE.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_WRONG_TRUST.xml
- EIDAS-Node/src/test/resources/SignModule_Service.xml
- EIDAS-Node/src/test/resources/SignModule_ServiceWithAlgorithmWhitelist.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_Conf1.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_Conf2.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_Conf3.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_Conf4.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_Conf5.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_Conf6.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_DOM-test.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_EidasOnly.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_Metadata.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_MetadataO.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_Service.xml

23. CertificateValidator
    a. the CHECK_VALIDITY_PERIOD_PROPERTY value was changed with the new configuration property name for "check.certificate.validity.period".
    b. the DISALLOW_SELF_SIGNED_CERTIFICATE_PROPERTY value was changed with the new configuration property name for "disallow.self.signed.certificate".
24. ProtocolSignerTest
    a. Property used in a map for tests was updated for the new configuration property name of "check.certificate.validity.period".
    b. the "disallow_self_signed_certificate" property used in a map of properties was updated by "disallow.self.signed.certificate" property.
    c. Add a key-value for signature.algorithm in the signingProperties map in the method to get the RSA signer.
25. SAMLEngineUtilsTest
    a. Property used in a map for tests was updated for the new configuration property name of "check.certificate.validity.period".
    b. the "disallow_self_signed_certificate" property used in a map of properties was updated by "disallow.self.signed.certificate" property.
    c. Add a key-value for signature.algorithm in the signingProperties map in the method to get the RSA signer.
26. EncryptionConfiguration
    a. New getter method to retrieve encryption algorithm whitelist as an ImmutableSet
27. EidasEncryptionConstants
    a. New class to provide a list of the encryption algorithms allowed by the 1.2 specifications
28. ProtocolCipherI
    a. Add new method to the interface to return the EncryptionConfiguration
29. AbstractProtocolCipher
    a. Modify class implementation to add the new method of the ProtocolCipherI interface
30. AbstractSamlEngineEncryption
    a. Modify class implementation to add the new method of the ProtocolCipherI interface

31. KeyStoreEncryptionConfigurator
    a. Update class to limit the "encryption.algorithm.whitelist" property to only the value allowed by the specifications

3.8.2 Configuration changes

The following entries were removed from external configuration files (e.g. eidas.xml, SignModule_Connector.xml, SignModule_Connector_EC.xml SignModule_Service.xml, SignModule_Service_EC.xml, EncryptModule_Connector.xml, EncryptModule_Service.xml) and moved into /EIDAS-SAMLEngine/src/main/resources/default/ defaultSamlEngineProperties.xml:

- request.sign.with.key.value
- response.sign.with.key.value
- response.encryption.mandatory
- response.sign.assertions
- assertion.encrypt.with.key.value
- enable.address.attribute.subject.confirmation.data
- signature.algorithm
- signature.algorithm.whitelist
- check_certificate_validity_period (**renamed to** check.certificate.validity.period)
- disallow_self_signed_certificate (**renamed to** disallow.self.signed.certificate)
- data.encryption.algorithm
- encryption.algorithm.whitelist
- key.encryption.algorithm
- key.encryption.algorithm.key.wrapping

A new keystore *"eidasKeyStore_METADATA_TC_EC"* was added to EIDAS-Config/keystore folder to provide an example of Trust chain with Elliptic Curve keys. The SamlEngine.xml, SignModule_Connector_EC.xml and SignModule_Service_EC.xml were modified to align with the new default signature algorithm. The keystore EIDAS-Config/keystore/eidasKeyStore.p12 and EIDAS-Config/keystore/eidasKeyStore_METADATA.p12 were updated with a new privateKeyEntry to replace the "local-demo-cert", and "metadata" which were outdated.

In the context of the changes related to the modification of the "check_certificate_validity_period" property to "check.certificate.validity.period" and of the "disallow_self_signed_certificate" to "disallow.self.signed.certificate", the following config files were modified :

- EIDAS-Config/server/eidas.xml
- EIDAS-Config/server/EncryptModule_Connector.xml
- EIDAS-Config/server/EncryptModule_Service.xml
- EIDAS-Config/server/SignModule_Connector.xml
- EIDAS-Config/server/SignModule_Connector_EC.xml
- EIDAS-Config/server/SignModule_Service.xml
- EIDAS-Config/server/SignModule_Service_EC.xml

The "check_certificate_validity_period" config entry was removed from these files and for encrypt module files the responseDecryptionIssuer and the SerialNumber were also modified. The "disallow_self_signed_certificate" was replaced by the new one.

3.9 ConfigurationSecurityBean code cleaning

Some properties in the ConfigurationSecurityBean were never used in the code, therefore it was decided to remove the support to these properties in order to clean the code.

3.9.1 Code changes

The properties "isMoaActive" and "cspFallbackCheck" (and their getters and setters) were removed from the ConfigurationSecurityBean class. The SecurityResponseHeaderHelper was also modify to not log the value of the isMoaActive property since it was used by the code. The applicationContext.xml was also updated to not provide values for these removed properties in the bean of ConfigurationSecurityBean type.

3.9.2 Configuration change

The support for the following properties was dropped :

- isMoaActive
- security.header.CSP.fallbackCheckMode

3.10 Replace JKS keystores by PKCS12 keystores

Since the JKS keystores are using a proprietary format. It was decided to migrate the keystores to the PKCS12 industry format since it is an standard format.

Therefore all keystores inside the sources (EIDAS-Config, EIDAS-Encryption, EIDAS-Node and EIDAS-SAMLEngine) were converted into PKCS12 keystores.

Previous keystores	New keystores
EIDAS-Config/keystore/eidasKeyStore_METADATA_TC_EC.jks	EIDAS-Config/keystore/eidasKeyStore_METADATA_TC_EC.p12
EIDAS-Config/keystore/demoKeys.jks EIDAS-Config/keystore/eidasKeyStore.jks	demoKeys keystore has been removed. EIDAS-Config/keystore/eidasKeyStore.p12
EIDAS-Config/keystore/eidasKeyStore_Service_CA.jks EIDAS-Config/keystore/eidasKeyStore_Service_CB.jks EIDAS-Config/keystore/eidasKeyStore_Service_CC.jks EIDAS-Config/keystore/eidasKeyStore_Service_CD.jks EIDAS-Config/keystore/eidasKeyStore_Service_CF.jks	Specific Country code Keystores have been removed since they weren't not used anymore
EIDAS-Config/keystore/eidasKeyStore_Connector_CA.jks EIDAS-Config/keystore/eidasKeyStore_Connector_CB.jks EIDAS-Config/keystore/eidasKeyStore_Connector_CC.jks EIDAS-Config/keystore/eidasKeyStore_Connector_CD.jks EIDAS-Config/keystore/eidasKeyStore_Connector_CF.jks	Specific Country code Keystores have been removed since they weren't not used anymore

Previous keystores	**New keystores**
EIDAS-Config/keystore/eidasKeyStore_METADATA.jks EIDAS-Config/keystore/eidasKeyStore_METADATA_EC.jks EIDAS-Config/keystore/eidasKeyStore_METADATA_TC.jks	EIDAS-Config/keystore/eidasKeyStore_METADATA.p12 EIDAS-Config/keystore/eidasKeyStore_METADATA_EC.p12 EIDAS-Config/keystore/eidasKeyStore_METADATA_TC.p12
EIDAS-Config/server/ignite/KeyStore/server.jks EIDAS-Config/server/ignite/KeyStore/trust.jks	EIDAS-Config/server/ignite/KeyStore/server.p12 EIDAS-Config/server/ignite/KeyStore/trust.p12
EIDAS-Encryption/src/test/resources/keystores/test.jks	EIDAS-Encryption/src/test/resources/keystores/test.p12

Previous keystores	New keystores
EIDAS-Node/src/test/resources/demoKeys.jks	demoKeys keystore has been removed
EIDAS-Node/src/test/resources/keystore/eidasKeyStore_Service_CA.jks	EIDAS-Node/src/test/resources/keystore/eidasKeyStore_Service_CA.p12
EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA.jks	EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA.p12
EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_EMPTY_TRUST.jks	EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_EMPTY_TRUST.p12
EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_WRONG_TRUST.jks	EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_WRONG_TRUST.p12
EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_NODE_CERT_TRUST.jks	EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_NODE_CERT_TRUST.p12
EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_ROOT_CA_TRUST.jks	EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_ROOT_CA_TRUST.p12
EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_SINGLE_CERTIFICATE.jks	EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_SINGLE_CERTIFICATE.p12
EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_INTERMEDIATE_CA_ROOT_CA_TRUST.jks	EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_INTERMEDIATE_CA_ROOT_CA_TRUST.p12
EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_INTERMEDIATE_TRUST_CA.jks	EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_INTERMEDIATE_TRUST_CA.p12
EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_TC.jks	EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_TC.p12
EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_TC_WITHOUT_INTERMEDIATE_CA_CERTIFICATE.jks	EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_TC_WITHOUT_INTERMEDIATE_CA_CERTIFICATE.p12
EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_TC_WRONG_ORDER_EXTRA_CERTIFICATE.jks	EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_TC_WRONG_ORDER_EXTRA_CERTIFICATE.p12
EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_TC_WITHOUT_ROOT_CA_CERTIFICATE.jks	EIDAS-Node/src/test/resources/keystore/eidasKeyStore_METADATA_TC_WITHOUT_ROOT_CA_CERTIFICATE.p12

Previous keystores	New keystores
EIDAS-Node/src/test/resources/keystore/eIDASkeystores/metadata_TC_WITHOUT_ROOT_CA.jks EIDAS-Node/src/test/resources/keystore/eIDASkeystores/metadata_TC_WITHOUT_INTERMEDIATE_CA_CERTIFICATE.jks EIDAS-Node/src/test/resources/keystore/eIDASkeystores/eidasKeyStore_Connector_CC_TrustedRoot.jks EIDAS-Node/src/test/resources/keystore/eIDASkeystores/eidasKeyStore_Connector_CC_TrustedIntermediate.jks EIDAS-Node/src/test/resources/keystore/eIDASkeystores/eidasKeyStore_Connector_CC_TrustedLeaf.jks EIDAS-Node/src/test/resources/keystore/eIDASkeystores/eidasKeyStore_Connector_CC_TrustedIntermediateTrustedRoot.jks EIDAS-Node/src/test/resources/keystore/eIDASkeystores/metadata.jks EIDAS-Node/src/test/resources/keystore/eIDASkeystores/metadataSingleCertificate.jks EIDAS-Node/src/test/resources/keystore/eIDASkeystores/metadataPlusExtraCertificate.jks	EIDAS-Node/src/test/resources/keystore/eIDASkeystores/metadata_TC_WITHOUT_ROOT_CA.p12 EIDAS-Node/src/test/resources/keystore/eIDASkeystores/metadata_TC_WITHOUT_INTERMEDIATE_CA_CERTIFICATE.p12 EIDAS-Node/src/test/resources/keystore/eIDASkeystores/eidasKeyStore_Connector_CC_TrustedRoot.p12 EIDAS-Node/src/test/resources/keystore/eIDASkeystores/eidasKeyStore_Connector_CC_TrustedIntermediate.p12 EIDAS-Node/src/test/resources/keystore/eIDASkeystores/eidasKeyStore_Connector_CC_TrustedLeaf.p12 EIDAS-Node/src/test/resources/keystore/eIDASkeystores/eidasKeyStore_Connector_CC_TrustedIntermediateTrustedRoot.p12 EIDAS-Node/src/test/resources/keystore/eIDASkeystores/metadata.p12 EIDAS-Node/src/test/resources/keystore/eIDASkeystores/metadataSingleCertificate.p12 EIDAS-Node/src/test/resources/keystore/eIDASkeystores/metadataPlusExtraCertificate.p12

Previous keystores	New keystores
EIDAS-SAMLEngine/target/test-classes/signatureTestKeystore.jks	EIDAS-SAMLEngine/target/test-classes/signatureTestKeystore.p12
EIDAS-SAMLEngine/target/test-classes/keyStoreMetadata.jks	EIDAS-SAMLEngine/target/test-classes/keyStoreMetadata.p12
EIDAS-SAMLEngine/target/test-classes/keyStoreCountry3.jks	EIDAS-SAMLEngine/target/test-classes/keyStoreCountry3.p12
EIDAS-SAMLEngine/target/test-classes/keyStoreCountry2.jks	EIDAS-SAMLEngine/target/test-classes/keyStoreCountry2.p12
EIDAS-SAMLEngine/target/test-classes/keyStoreCountry0.jks	EIDAS-SAMLEngine/target/test-classes/keyStoreCountry0.p12
EIDAS-SAMLEngine/target/test-classes/keyStoreWithMultiplePrivateKeyEntries.pjks	EIDAS-SAMLEngine/target/test-classes/keyStoreWithMultiplePrivateKeyEntries.p12
EIDAS-SAMLEngine/target/test-classes/keyStoreCountry1.jks	EIDAS-SAMLEngine/target/test-classes/keyStoreCountry1.p12
EIDAS-SAMLEngine/src/test/resources/signatureTestKeystore.jks	EIDAS-SAMLEngine/src/test/resources/signatureTestKeystore.p12
EIDAS-SAMLEngine/src/test/resources/keyStoreMetadata.jks	EIDAS-SAMLEngine/src/test/resources/keyStoreMetadata.p12
EIDAS-SAMLEngine/src/test/resources/keyStoreCountry3.jks	EIDAS-SAMLEngine/src/test/resources/keyStoreCountry3.p12
EIDAS-SAMLEngine/src/test/resources/keyStoreCountry2.jks	EIDAS-SAMLEngine/src/test/resources/keyStoreCountry2.p12
EIDAS-SAMLEngine/src/test/resources/keyStoreCountry0.jks	EIDAS-SAMLEngine/src/test/resources/keyStoreCountry0.p12
EIDAS-SAMLEngine/src/test/resources/keyStoreWithMultiplePrivateKeyEntries.jks	EIDAS-SAMLEngine/src/test/resources/keyStoreWithMultiplePrivateKeyEntries.p12
EIDAS-SAMLEngine/src/test/resources/keyStoreCountry1.jks	EIDAS-SAMLEngine/src/test/resources/keyStoreCountry1.p12

Along the conversion of the keystores, some tests classes and configuration files needed to be modified to use the new keystores.

Here is the list of modified files :

- EIDAS-Encryption/src/test/java/eu/eidas/encryption/SAMLAuthnResponseDecrypterTest.java
- EIDAS-Node/src/test/resources/EncryptModule_Service.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_EMPTY_TRUST.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_INTERMEDIATE_CA_ROOT_CA_TRUST.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_INTERMEDIATE_CA_TRUST.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_NODE_CERT_TRUST.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_ROOT_CA_TRUST.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_SINGLE_CERTIFICATE.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_TC.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_TC_WITHOUT_INTERMEDIATE_CA_CERTIFICATE.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_TC_WITHOUT_ROOT_CA_CERTIFICATE.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_TC_WRONG_ORDER_EXTRA_CERTIFICATE.xml
- EIDAS-Node/src/test/resources/SignModule_METADATA_WRONG_TRUST.xml
- EIDAS-Node/src/test/resources/SignModule_Service.xml
- EIDAS-Node/src/test/resources/SignModule_ServiceWithAlgorithmWhitelist.xml
- EIDAS-SAMLEngine/src/test/java/eu/eidas/auth/engine/core/impl/AbstractProtocolDecrypterTest.java

- EIDAS-SAMLEngine/src/test/java/eu/eidas/auth/engine/core/impl/AbstractProtocolEncrypterTest.java
- EIDAS-SAMLEngine/src/test/java/eu/eidas/auth/engine/core/impl/AbstractProtocolSignerTest.java
- EIDAS-SAMLEngine/src/test/java/eu/eidas/auth/engine/core/impl/ProtocolSignerTest.java
- EIDAS-SAMLEngine/src/test/java/eu/eidas/auth/engine/xml/opensaml/SAMLEngineUtilsTest.java
- EIDAS-SAMLEngine/src/test/java/eu/eidas/engine/metadata/TestMDGenerator.java
- EIDAS-SAMLEngine/src/test/resources/EncryptModule_DOM-test_empty.xml
- EIDAS-SAMLEngine/src/test/resources/EncryptModule_DOM-test_false.xml
- EIDAS-SAMLEngine/src/test/resources/EncryptModule_DOM-test_true.xml
- EIDAS-SAMLEngine/src/test/resources/EncryptModule_Metadata.xml
- EIDAS-SAMLEngine/src/test/resources/EncryptModule_MetadataO.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_Conf1.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_Conf2.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_Conf3.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_Conf4.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_Conf5.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_Conf6.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_DOM-test.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_EidasOnly.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_Metadata.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_MetadataO.xml
- EIDAS-SAMLEngine/src/test/resources/SignModule_Service.xml

Modifications to these files are to update the KeystorePath and the KeystoreType.

3.11 Disabled support for TLSv1.0 & TLSv1.1 in Java 8 revision 291 and Java 11 revision 11

In releases 291 of Java 8 (no longer supported) and 11 of Java 11, support for TLSv1.0 and TLSv1.1 has been disabled by the publisher for security reasons.

3.11.1 Code changes

In HTTPSConnectionWithEIDASCipherSuiteTest.java:

- Test TLS_1ClientCS_TLS_1ServerCS_RSA_1024 was adapted to only run for java versions before 1.8.0_291 and 11.0.11, and was renamed to testHttpClientWithTLS1_1IfSupportedByJdkAndAlgorithmRSA1024
- Test TLS_1ClientCS_TLS_1ServerCS_EC_256 was adapted to only run for java versions before 1.8.0_291 and 11.0.11, and was renamed to testHttpClientWithTLS1_1IfSupportedByJdkAndAlgorithmEC256
- Added test testHttpClientWithTLS1_1IfNotSupportedByJdkAndAlgorithmRSA1024 which expects an exception when running for java versions after 1.8.0_291 and 11.0.11
- Added test testHttpClientWithTLS1_1IfNotSupportedByJdkAndAlgorithmEC256 which expects an exception when running for java versions after 1.8.0_291 and 11.0.11

3.12 New metadata signature algorithm configuration entry

In order to let the possibility to sign the request/response and the metadata with a different signature algorithm a new configuration key entry was added.

3.12.1 Code changes

The *SigningContext* class was added to represent the context information needed to sign and also a corresponding test class *SigningContextTest*. The *createCredential* method from the *CertificateUtil* class was modified to handle null private key entry. A default value for the new metadata.signature.algorithm configuration key entry was added to the *defaultSamlEngineProperties.xml* file. The *AbstractProtocolSigner* class implementation was modified to hold 3 SigningContext instead of credentials and boolean variables, the *sign* methods were also modified, deprecating the method *sign(SignableObject, boolean)* which should be replaced by the new *sign(SignableObject, SigningContext).* The *getSignatureConfiguration* method from the *KeyStoreSignatureConfigurator* class was updated so it also fetch the metadata signature algorithm parameter. The *SignatureConfiguration* class was modified to add a Builder and deprecated constructors. The METADATA_SIGNATURE_ALGORITHM key was added in the *SignatureKey* class. The SignModule test configuration files were also modified to also include the metadata signature algorithm. In ProtocolSignerI the following methods were deprecated: getPublicSigningCredential, isRequestSignWithKey, isResponseSignWithKey. In MetadataSignerI the following method was deprecated getPublicMetadataSigningCredential.

3.12.2 Configuration changes

The "metadata.signature.algorithm" configuration key entry was added and can be configured in the SignModule_Connector.xml or SignModule_Service.xml files. The default value for this configuration entry is "*http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha512*"

3.13 New key encryption Agreement Method algorithm configuration entry

In order to configure the Agreement method to be used when encrypting with Key Agreement, a new configuration entry was added.

3.13.1 Code changes

Added a constant "*KEY_ENCRYPTION_AGREEMENT_METHOD_ALGORITHM*" for the "*key.encryption.agreement.method.algorithm*" configuration entry in the EncryptionKey. Added a default configuration entry for the Agreement method algorithm in the defaultSamlEngineProperties.xml. Modified EncryptionConfiguration class to add a builder and an instance variable for the Agreement method. Modified the KeyStoreEncryptionConfigurator to get the value of the Agreement method algorithm and to use the EncryptionConfiguration builder instead of the constructors. Deprecated EncryptionConfiguration constructor that is missing keyEncryptionAgreementMethodAlgorithm in favor of EncryptionConfiguration.Builder. Deprecated constructors in AbstractProtocolCipher AbstractProtocolDecrypter AbstractProtocolEncrypter BaseProtocolDecrypter BaseProtocolEncrypter in favor of constructor with parameter of EncryptionConfiguration. Added an instance variable in SAMLAuthnResponseEncrypter to configure Agreement method to use when using key agreement. Modified the AbstractProtocolEncrypter to use the Agreement method algorithm

3.13.2 Configuration changes

The "key.encryption.agreement.method.algorithm" configuration entry was added. The default value for this configuration entry is "*http://www.w3.org/2009/xmlenc11#ECDH-EC*"

3.14 Other changes requiring no action

3.14.1 PTES-012 - INFO - Traffic analysis

In order to prevent indexation of the node pages, the X-Robots-Tag (noindex, nofollow) was added to the response headers. This change was done in class SecurityResponseHeaderHelper.java

3.14.2 CVE-2020-29245 OWASP Encoder version update

The version of the org.owasp.encoder:encoder and org.owasp.encoder:encoder-jsp librairies have been updated to the most recent version (v1.2.3).

3.14.3 PTES-011 CSP Report URL Poisoning

In order to stop the CSP report servlet from responding when security.header.CSP.enabled is set to false, the servlet will now return a 404 not found HTTP error instead of the 403 forbidden.

3.14.4 SAST-ERR-004

In the engine module BaseMetadataFetcher was updated to throw an EIDASMetadataProviderException upon encountering non-uri metadata locations to prevent the injection of characters that could cause forging in the logs. Similarly WhitelistingMetadataFetcher throws an EIDASMetadataRuntimeException.

3.14.5 IEIDASLogger could be removed (EID-1126)

Since eu.eidas.auth.commons.IEIDASLogger interface interface is not used anymore it has been removed along with its references.

3.14.6 Remove EIDAS-ConfigModule as dependency from node and demo tools

Module "eu.eidas:eidas-configmodule" has been removed since it is not needed.

The references and dependencies on that module have been removed from the following pom.xml :

- EIDAS-Metadata/pom.xml
- EIDAS-SAMLEngine/pom.xml
- EIDAS-Node/pom.xml
- EIDAS-Parent/pom.xml

The removal of the dependency has originated the following changes:

- In the EIDAS-Metadata module
    - the FileMetadataLoader.java has been modified to replace the use of the FileService by the java 8, Files and Paths classes.
- In the EIDAS-SAMLEngine module,
    - the ConfigurationAdapter.java was removed
    - the ProtocolEngineFactory.java has been modify with the removal of the methods using a parameter types defined in the EIDAS-ConfigModule, such as :

```
public static ProtocolEngineI createProtocolEngine(String instanceName,
CertificateConfigurationManager configManager, ProtocolProcessorI protocolProcessor,
SamlEngineClock samlEngineClock) throws ProtocolEngineConfigurationException
```

  - the ProtocolEngineConfigurationFactory has been modify with a removal of all methods using parameters types defined in the EIDAS-ConfigModule, such as:

    ```
    public ProtocolEngineConfigurationFactory(CertificateConfigurationManager configManager)
    throws ProtocolEngineConfigurationException
    public static ImmutableMap<String, ProtocolEngineConfiguration>
    getConfigurationMap(CertificateConfigurationManager configManager) throws
     ProtocolEngineConfigurationException
    ```

    and some other private methods.
- In the EIDAS-Node module,
  - the TestEidasNodeMetadataLoader was also modified

3.14.7 Replace generic exceptions by more specific exceptions

In the context of code improvement, some methods are throwing generic exceptions like RuntimeException. This makes it difficult for callers to perform proper error handling and error recovery. Therefore, the RuntimeException have been replace by more specific exception to help handle each specific circumstance.

Here is a list of the changes :

- Introduction of UnsupportedTypeException which is an extension of RuntimeException which has been added to the LightCommons module
- CertificateUtil#getCountry(KeyInfo keyInfo) *throws CertificateException* (now throws eu.eidas.encryption.exception.CertificateException)
- EidasProtocolProcessor to handle the update CertificateUtil#getCountry method
- HashAndCounterGenerator constructor has been modified to throw an InvalidParameterEidasException instead of RuntimeException when hashAlgorithm is not valid
- VarMap#map(Type type) method was modified to throw a UnsupportedTypeException instead of RuntimeException when the type is not handled.
- GenericTypeReflector#getExactDirectSuperTypes(Type type) and GenericTypeReflector#erase(Type type) methods were both modify to throw UnsupportedTypeException instead of RuntimeException

3.14.8 Replace catch of generic exceptions by catch of more specific exceptions

In the context of code improvement, multiple classes were modified to not handle generic exception like "Exception" or "RuntimeException" which may obfuscate the real error.

Therefore, all the following classes have modified to handle more specific exceptions instead:

- EIDAS-Commons/src/main/java/eu/eidas/auth/commons/DateUtil.java
- EIDAS-Commons/src/main/java/eu/eidas/auth/commons/EidasDigestUtil.java
- EIDAS-Commons/src/main/java/eu/eidas/auth/commons/tx/LightTokenEncoder.java
- EIDAS-Commons/src/main/java/eu/eidas/auth/commons/lang/reflect/ReflectionUtil.java
- EIDAS-Encryption/src/main/java/eu/eidas/auth/commons/xml/opensaml/OpenSamlHelper.java
- EIDAS-Light-Commons/src/main/java/eu/eidas/auth/commons/attribute/AttributeDefinition.java
- EIDAS-Metadata/src/main/java/eu/eidas/auth/engine/metadata/impl/BaseMetadataFetcher.java
- EIDAS-Metadata/src/main/java/eu/eidas/auth/engine/metadata/EidasMetadata.java

- EIDAS-Node/src/main/java/eu/eidas/node/security/ContentSecurityPolicyFilter.java
- EIDAS-Node/src/main/java/eu/eidas/node/security/RemoveHttpHeadersFilter.java
- EIDAS-Node/src/main/java/eu/eidas/node/utils/EidasNodeErrorUtil.java
- EIDAS-Node/src/main/java/eu/eidas/node/InternalExceptionHandlerServlet.java
- EIDAS-Node/src/main/java/eu/eidas/node/connector/ConnectorExceptionHandlerServlet.java
- EIDAS-Node/src/main/java/eu/eidas/node/service/ServiceExceptionHandlerServlet.java
- EIDAS-Node/src/main/java/eu/eidas/node/service/ColleagueRequestServlet.java
- EIDAS-Node/src/main/java/eu/eidas/node/logging/MessageLoggerUtils.java
- EIDAS-Node/src/main/java/eu/eidas/node/logging/connector/messages/ConnectorIncomingEidasResponseLogger.java
- EIDAS-Node/src/main/java/eu/eidas/node/logging/connector/messages/ConnectorOutgoingEidasRequestLogger.java
- EIDAS-Node/src/main/java/eu/eidas/node/logging/connector/messages/ConnectorIncomingLightRequestLogger.java
- EIDAS-Node/src/main/java/eu/eidas/node/logging/connector/messages/ConnectorOutgoingLightResponseLogger.java
- EIDAS-Node/src/main/java/eu/eidas/node/logging/service/messages/ProxyServiceIncomingEidasRequestLogger.java
- EIDAS-Node/src/main/java/eu/eidas/node/logging/service/messages/ProxyServiceOutgoingEidasResponseLogger.java
- EIDAS-Node/src/main/java/eu/eidas/node/logging/service/messages/ProxyServiceIncomingLightResponseLogger.java
- EIDAS-Node/src/main/java/eu/eidas/node/logging/service/messages/ProxyServiceOutgoingLightRequestLogger.java
- EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/core/impl/AbstractProtocolCipher.java
- EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/core/impl/AbstractProtocolDecrypter.java
- EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/core/impl/AbstractProtocolEncrypter.java
- EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/core/impl/AbstractProtocolSigner.java
- EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/core/DefaultCoreProperties.java
- EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/configuration/dom/DefaultProtocolEngineConfigurationFactory.java
- EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/DefaultProtocolEngineFactory.java
- EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/configuration/dom/DOMConfigurator.java
- EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/configuration/dom/KeyStoreConfigurator.java
- EIDAS-SpecificCommunicationDefinition/src/main/java/eu/eidas/specificcommunication/protocol/util/LightMessagesConverter.java

The two following test classes have also been modified:

- EIDAS-Node/src/test/java/eu/eidas/node/logging/connector/messages/ConnectorOutgoingEidasRequestFullLoggerTest
- EIDAS-Node/src/test/java/eu/eidas/node/logging/service/messages/ProxyServiceIncomingEidasRequestFullLoggerTest

3.14.9 Enabling/Disabling of security.header.CSP.enabled not working

A fix was implemented to make the entry security.header.CSP.enabled value to enable/disable the inclusion of the following headers: Content-Security-Policy, *X-Content-Security-Poli*cy and *X-WebKit-CSP.*

3.14.10 *SecurityRequestFilter is being activated for ServiceProvider instead of SpecificConnectorRequest*

At web.xml, the filter configuration for SecurityRequestFilter was changed and SpecificConnectorRequest was added at the param-value so that the Security filter is also activated for request arriving at SpecificConnectorRequest servlet. The ServiceProvider servlet was kept in the list of included servlets so that the Security filter for requests arriving at ServiceProvider is activated in the case of monolithic deployment. The logging message for exceeded number of requests per address was adapted in order to be more accurate.

3.14.11 SecurityRequestFilter is not being activated for IdpResponse and SpecificProxyServiceResponse

At web.xml the list of includedServlets for SecurityRequestFilter was appended with SpecificProxyServiceResponse,IdpResponse so that the Security filter for requests arriving at the SpecificProxyServiceResponse or IdpResponse servlets are activated. The IdpResponse servlet was added in the list of included servlets so that the Security filter for requests arriving at IdpResponse is activated in the case of monolithic deployment.

3.14.12 Connector sends SAML request when active.module.connector is false

A regression introduced in 2.5 was fixed that caused the connector to continue accepting and processing an authentication requests when the active.module.connector is set to false.

3.14.13 Deployment of standalone eIDAS node should not require specific connector and proxy configuration

In order to fix the deployement of the eIDAS node without both the SPECIFIC_CONNECTOR_CONFIG_REPOSITORY and SPECIFIC_PROXY_SERVICE_CONFIG_REPOSITORY environment parameter the following files have been modified :

- specificCommunicationDefinitionApplicationContext.xml
- specificCommunicationDefinitionEnvironmentContext.xml

This implementation addresses EID-1189 and EID-1191.

3.14.14 Sometimes the node fails to block the requests from Specific Connector

In order to fix the method validating the maximum number of requests in a given timeframe, the checkRequest method in AbstractSecurityRequest.java has been modified

3.14.15 Regression 2.6 : Error page has changed (http Error 500) - Country Code

Detailed error page messages concerning countrycode validation of various identifiers in the SAML Response now return "202003 - Invalid SAML Response token.". Details can be found in the logs.

3.14.16 Remove validation of 2 maximum number of MDSs

As clarified in the technical subgroup of March 2021, the number of Minimum data set doesn't need to be validated. Therefore, method checkRepresentationResponse in class EidasProtocolProcessor.java was changed and returns true, so no validation is applied. Furthermore, the tests testCheckRepresentationResponseInvalidTooManyMDS and testCheckRepresentationResponseInvalidTooFewMDS in EidasProtocolProcessorTest.java were removed and Javadoc of checkRepresentationResponse in both EidasProtocolProcessor.java and ProtocolProcessorI.java were updated. This implementation addresses EID-1188.

3.14.17 Junit tests coverage improvements

As a part of the Error Handling improvements, code coverage was increased on the following classes:

- EidasRequestedAttributeValidator: Added EidasRequestedAttributeValidatorTest class with methods 'testValidateOpenSaml', 'testValidate', 'testValidateExceptionNameIsRequired' and 'testValidateExceptionNameFormatIsRequired'.
- EidasRequestedAuthContextValidator: Added 'failOnNoLevelsOfAssurance' test to EidasRequestedAuthContextValidatorTest.
- AUSERVICECitizen: Added 'testCheckMandatoryAttributes' test to AUSERVICECitizenTestCase.
- GenderAttributeValueMarshaller: Added GenderAttributeValueMarshallerTest class with method 'testUnmarshalWithGenderValueNull".
- AttributeDefinition: Improved AttributeDefinitionTest with methods "testBuilderValidateInvalidNameUri", "testBuilderValidateRelativeNameUri", "testBuilderValidateRelativeXmlNamespaceUri" and "testBuilderValidateInvalidXmlNamespaceUri".
- AttributeValidator: Added 'testMarshallAttributeHandlesAttributeValueMarshallingException' test to AttributeValidatorTest.
- EidasNodeValidationUtil: Added 'testValidateAssertionConsumerURLWithInvalidMetadataAssertionConsumerURL', 'testValidateBindingWithBindingMethodNull', 'testValidateServiceDestinationWithInvalidServicePostLocation', 'testValidateServiceDestinationWithInvalidServiceRedirectLocation' tests to EidasNodeValidationUtilTestCase.
- ProtocolEngineFactory: Added 'testCreateProtocolEngineAccessorIOException' and 'testConstructorAccessorIOException' to ProtocolEngineFactoryTest.
- SAMLEngineUtils: Added 'testGetValidIssuerValueWithValueNull' and testGetValidIssuerValueWithValueValidScheme test to SAMLEngineUtilsTest.
- IncomingLightRequestValidatorLoaComponent: Added 'failOnDocumentBuilderIOException' to IncomingLightRequestValidatorLoAComponentTest.java.
- AssertionUtil: Improved AssertionUtilTest with methods 'testGetNameIdWithMissingNameIdFormatInRequest' and 'testGetNameIdWithMissingNameIdFormatOtherSamlFormat'.
- AbstractProtocolCipher: added testValidateEncryptionAlforithmFail to AbstractProtocolCipherTest
- ExtensionsSchemaValidator: Added ExtensionsSchemaValidatorTest with methods 'testValidateWithUnknownXMLObjectsNull' and 'testValidateWithElementNotInstanceOfXSAnyImpl'
- LightTokenEncoder: added testDecodeFailsTokenTooLarge and testDecodeFailsIdBlank to LightTokenEncoderTestcheckRequest
- CertificateValidator: Added CertificateValidatorTest with methods 'testCheckCertificateValidityPeriod', 'testCheckCertificateValidityPeriodWithCheckedValidityPeriod', 'testCheckCertificateIssuerWithCertificateSigned', 'testCheckCertificateIssuerWithCertificateNotSigned', 'testCheckCertificateIssuerWithPropertiesParameter' and 'testCheckCertificateValidityPeriodWithPropertiesParameter'

- AUSERVICE: Added 'testValidateIdpResponseWithIdpResponseNull', 'testValidateIdpResponseWithMissingMandatoryAttributes' and 'testValidateIdpResponseWithMissingMandatoryAttributeSet' tests to AUSERVICETestCase
- ResponseUtil: Added 'testVerifyTimeConditionsWithNotBeforeNull', 'testVerifyTimeConditionsWithNotOnOrAfterNull', 'testFindAttributeStatementWithAssertionNull', testVerifyTimeConditionsWhenNotBeforeIsAfterNowPlus1Minute, testVerifyTimeConditionsWhenNotOnOrAfterIsBeforeNow, testFindAttributeStatementWithAssertionWithAttributeStatement, testFindAttributeStatementWithAssertionWithXMLObjectButNotAttributeStatement tests to ResponseUtilTest
- AbstractProtocolDecrypter: Added testDecryptResponseNotEncrypted and testDecryptResponseIncorrectAssertions to AbstractProtocolDecrypterTest
- AbstractProtocolEncrypter: added AbstractProtocolEncrypterTest with tests testEncryptSamlResponseDestinationCertificateNull and testEncryptSamlResponseInvalidAssertions
- AttributeRegistry: added testLogRetrievedAttributesTrace, testLogRetrievedAttributesIoException, testGetAttributesIoException and testGetAttributesGetByFilterIoException to class AttributeRegistryTest
- BuilderFactoryUtil: Added BuilderFactoryUtilTest with 'testBuildXmlObjectWithBuilderNull', 'testBuildXmlObjectWithClassParameter' and 'testBuildXmlObjectWithTwoQnameParameters' tests
- LightMessagesConverter: Added class LightMessagesConverterTest with tests testGetByName, testGetByNameWhenNameUriNull, testGetByNameWhenDefinitionNameUriNull, testGetByNameWhenDefinitionNotPresent
- EidasDigestUtil: added class EidasDigestUtilTest with tests testHashBytes, testHashNullProvider, testHashNoSuchProvider, testHashNoSuchAltorithm and testHashPersonalToken
- MetadataUtil: added tests testConvertKeyDescriptorsSigningCertificateException and testConvertKeyDescriptorEncryptionCertificateException to MetadataUtilTest
- FileMetadataLoader: added test testGetEntityDescriptorsJavaCharsInID to class TestEidasNodeFileMetadataProcessor
- AUCONNECTOR: Added 'testPrepareEidasRequest', 'testPrepareEidasRequestWithAssertionConsumerServiceURLAndBinding', 'testValidateAuthenticationRequestSPTypeWithDifferentSpType', 'testValidateAuthenticationRequestSPTypeWithNoSpType', 'testGetAuthenticationResponse', 'testGetAuthenticationResponseWithNoAttributes' to class AUCONNECTORTestCase. Updated "runBeforeClass", "testGetAuthenticationRequest", "testPrepareEidasRequestWithPrivateConnectorPrivateRequestSPType", "testPrepEIDAS-Node/src/test/java/eu/eidas/node/logging/service/messages/ProxyServiceIncomingEidasRequestFullLoggerTestareEidasRequestWithPublicConnectorPublicRequestSPType", "testPrepareEidasRequestWithRepresentativeAttributes", "testPrepareEidasRequestWithUndefinedConnectorPublicRequestSPType", "testPrepareEidasRequestWithUndefinedConnectorPrivateRequestSPType", "testPrepareEidasRequestWithPublicConnectorUndefinedRequestSPType", "testPrepareEidasRequestWithPrivateConnectorUndefinedRequestSPType",· "testGetAuthenticationRequestWithPrivateConfigPublicRequestSpTypes", "testGetAuthenticationRequestWithPublicConfigPrivateRequestSpTypes" in class AUCONNECTORTestCase.
- ColleagueRequestServlet: added testBuildLightRequestIllegalArguments to class ColleagueRequestServletTest
- OpenSamlHelper: Added OpenSamlHelperTest with 'testUnmarshallFromDomWithDocumentNull', 'testUnmarshallFromDomWithRootNull', 'testUnmarshallFromDomWithUnmarshallerFactoryNull', 'testUnmarshallFromDomWithUnmarshallerNull' and 'testUnmarshallFromDomWhenExceptionIsThrown' tests
- AbstractCachingMetadataFetcher: Added AbstractCachingMetadataFetcherTest with 'getEidasMetadataFetchNewMetadata' and 'getEidasMetadataNoValidNewMetadata'.

- XmlSchemaUtil: Fixed and renamed testValidateSchema to testValidateSchemaDocumentMalicious, added testValidateSchemaString, testValidateSchemaStringMalicious and testValidateSamlSchemaString in class xmlSchemaUtilTest
- SecurityRequestFilter: Added SecurityRequestFilterTest with test methods doFilterServletNotFiltered, doFilterBypassValidation, doFilterBypassCspReportHandlerServlet, doFilterDomainIsNull, doFilterDomain, doFilterDomainRemoveTailingSlash, doFilterDomainInvalidSpUrl, doFilterDomainTrustedDomains, doFilterDomainNoDomains, doFilterRateLimitIP, doFilterRateLimitIPFourTimes, doFilterRateLimitIPForgivingFourTimes and doFilterRateLimitRefererDomainFourTimes
- SpecificConnectorRequestServlet: Added SpecificConnectorRequestServletTest with 'testGetLightRequest', 'testGetLightRequestWithLightRequestNull', 'testGetLightRequestWhenIllegalArgumentExceptionIsCaught' and 'testGetLightRequestWhenSpecificCommunicationExceptionIsCaught' tests
- AbstractSecurityRequest: Added class AbstractSecurityRequestTest with tests testCheckRequestKnownAddressBelowTreshold, testCheckRequestKnownAddressOverTreshold, testCheckRequestNewAddress, testCheckRequestKnownAddress, testCheckDomainKnown, testCheckDomainUnknown and testCheckDomainBadSpUrl
- AbstractProtocolEngine: Added AbstractProtocolEngineTest with 'getConfigurationWithProtocolEngineConfigurationNull', 'testUnmarshallWhenUnmarshallExceptionIsCaught', 'testMarshallWhenMarshallExceptionIsCaught', 'testCheckReceivingUnencryptedResponsesAllowed', 'testCheckReceivingUnencryptedResponsesAllowedWithEncryptionEnabled', 'testCheckSendingUnencryptedResponsesAllowed', 'testCheckSendingUnencryptedResponsesAllowedWithEncryptionEnabled', 'testSignResponse' and 'testSignResponseWithEncryptionEnabledAndCountryCodeNull'
- AUCONNECTORSAML: modified test processSpRequestWithActiveConnectorModule to cover the entire method, and added testProcessSpRequestNoLoaInRequestState, testProcessSpRequestErrorFetchingMetadata, testProcessProxyServiceResponseUnmarshallException, testProcessProxyServiceResponseWithAudienceRestriction, testProcessProxyServiceResponseFailureNoStatusMessageCode, testProcessProxyServiceResponseFailureHasStatusMessageCode, testProcessProxyServiceResponseEmptyCache, testProcessProxyServiceResponseBlankInResponseToId, testExtractErrorMessage, testExtractErrorMessageNotFound, testCheckIdentifierFormatLegalPerson, testCheckIdentifierFormatLegalPersonInvalid, testIsExpectedProxyServiceCountryCode, testGenerateAuthenticationRequtestNotInstanceOfEidasAuthenticationRequest, testGenerateAuthenticationRequest, testGenerateAuthenticationRequestErrorInEngine, testGenerateAuthenticationRequestEmptyMetadataUrl, testCheckAudienceRestriction, testValidateSamlCoreNameIdFormatIdentifier and testValidateSamlCoreNameIdFormatIdentifierUnknown to class AUCONNECTORSAMLTest
- AbstractProtocolSigner: Added 'getTrustedCertificateWhenCertificateIsUntrusted', 'testAddAllSignatureCertificatesToCredentialWhenEIDASSAMLEngineExceptionIsThrown', 'testCreateKeyInfoWhenOnlyKeyInfoNoCertIsTrue', 'testCreateKeyInfoWhenSecurityExceptionIsCaught', 'testSignWhenMarshallingExceptionIsCaught', 'testSignWhenSignatureExceptionIsCaught', 'testValidateSamlSignatureStructureWhenSignatureExceptionIsCaught', 'testValidateSignatureAlgorithmWhenEIDASSAMLEngineExceptionIsThrown' and 'testGetTrustedCertificateFromRSAKeyValueWhenEIDASSAMLEngineExceptionIsThrown' to class AbstractProtocolSignerTest
- AUSERVICESAML: Added 'testProcessIdpSpecificResponse', 'testProcessIdpSpecificResponseWithLevelOfAssuranceNull', 'testProcessIdpSpecificResponseWhenInternalErrorEIDASExceptionIsThrown', 'testGenerateErrorAuthenticationResponse', 'testGenerateErrorAuthenticationResponseWhenInternalErrorEIDASExceptionIsThrown', 'testProcessConnectorRequestWithLevelOfAssuranceNull', 'testProcessConnectorRequestWhenBindingValidationIsFalse', 'testProcessConnectorRequestWhenRequestIsUnsupported',

'testProcessConnectorRequestWithInvalidLoA', 'testProcessConnectorRequestWithColleagueRequestInvalidLoA', 'testProcessConnectorRequestWithRequestRepresentativeAttributeInvalid', 'testProcessConnectorRequestWithIssuerNull', 'testGetIssuerMetadataParametersWhenEIDASSAMLEngineExceptionIsThrown', 'testCheckAttributeListWithRequestedAttributesEmpty', 'testCheckAttributeList', 'testCheckAttributeListWithUnsupportedMandatoryAttributes', 'testCheckNameIDFormat', 'testCheckNameIDFormatWithInvalidNameIdFormat', 'testCheckCountryCode' and 'testCheckCountryCodeWithInvalidCountryCode' to class AUSERVICESAMLTest

- EidasProtocolProcessor: Added tests testUnmarshallRequestContainingXSStringValueErrorBuildingRequest, testUnmarshallRequestContainingXSSStringValueErrorUnmarshallingValue, testValidateRequestAgainstMetadataExceptionBuildingUpdatedRequest, testValidateParamResponseFailStatusCodeBlank, testValidateParamResponseFailIdBlank, testAddExtensionSPTypeBlank, testValidateRequestTypeInvalid, testValidateIdBlank, testValidateIssuer, testExtractLevelsOfAssuranceComparisonEmpty, testFillRequestedAttributesExceptionMarshalling, testFilterSupportedAttributeNames, testFilterSupportedAttributeNamesNotSupportedButRequired, testFilterSupportedAttributeNamesNotSupportedNotRequired, testFilterSupportedAttributeNamesNotModified, testCheckRequiredAttributeCompiles, testGetAttributeDefinitionNullable, testIsAcceptableHttpRequestMetadataFetcherNull, testIsAcceptableHttpRequestMetadataParametersNull, testIsAcceptableHttpRequestMismatchedAssertionConsumerServiceURL, testIsAcceptableHttpRequestNoSpType, testIsAcceptableHttpRequestInvalidBinding, testIsAcceptableHttpRequestSPTypeInRequestAndMetadata, testIsAcceptableHttpRequestExceptionInMetadataFetcher, testIsAcceptableHttpRequest, testMarshallRequestNotEidasAuthenticationRequest and performed code cleanup in class EidasProtocolProcessorTest
- ProtocolEngine: Added generateRequestMessage, generateRequestMessageNull, generateRequestMessageTryCatchEIDASSAMLEngineException, generateResponseMessage, generateResponseMessageSignAssertion, generateResponseMessageResponseSignAssertionsEnabledResponseDomNull, generateResponseMessageSignAssertionException, encryptAndSignAndMarshallResponseTryCatchEIDASSAMLEngineException, generateResponseErrorMessage, unmarshallRequest, unmarshallRequestMessageBytesNull, unmarshallRequestAndValidate, unmarshallRequestAndValidateRequestIsNull, unmarshallResponse, unmarshallResponseisNull, validateSignatureNoSignature, validateSignatureNullSignature, validateSignatureNoIssuer, validateSignatureTryCatchEidasSamlEngineException, validateRequestWithValidatorSuiteWrongIssuerNameIDFormat, validateResponseWithValidatorSuiteMultipleAssertions, validateSignatureAndDecryptAndValidateAssertionSignaturesValidateSignatureEnabledNoSignature, validateSignatureAndDecryptAndValidateAssertionSignaturesValidateSignatureDisabled, validateSignatureAndDecryptAndValidateAssertionSignaturesIssuerMissing, validateSignatureAndDecryptAndValidateAssertionSignatures, validateAssertionSignaturesWithTrustedCertificates, validateAssertionSignaturesTryCatchEIDASSAMLEngineException and updated unmarshallResponseAndValidate to ProtocolEngineTest.
- LightJAXBCodec: Added testMarshallUnmarshallBigResponse, testMarshallNullInput, testUnmarshallRequestNullInput, testUnmarshallResponseNullInput in class LightJAXBCodecTest.

3.14.18 Fix of metadata entity certificate chain publishing in metadata

The node used to publish in the metadata all the certificates present in the keystore where the key to sign the metadata was present, instead of just the certificate chain of the key. The AbstractProtocolSigner class was modified to not reverse the order of the entityCertificate chain from the "privateMetadataSigningCredential" and the "publicMetadataSigningCredential". The CertificateUtil class was added a method to create credential taking in a PrivateKeyEntry parameter allowing to use both the key itself and its entityCertificateChain to create the credential object. The previously "createCredential" method taking only a certficate and a privateKey as parameters was deprecated.

The TestEidasNodeMetadataTrustChain class was updated to allow some unit tests to be valid for adapting to the new conditions:

- test testValidMetadataSignatureWrongOrderExtraCertificateWrongTrust renamed to testValidMetadataSignatureExtraCertificateWhichDoesNotSignMetadataButTrusted
- testValidMetadataSignatureWithoutRootCaPublishedTrustingIntermediateCA, testValidMetadataSignatureWithoutRootCaPublishedTrustingRootCA, testValidMetadataSignatureWithoutIntermediateCAWithoutRootCAPublishedTrustingIntermediateCARootCA,

some other tests were removed since were not needed anymore:

- testValidMetadataSignatureWrongOrderExtraCertificate, testValidMetadataSignatureWithoutIntermediateCaPublishedTrustingRootCA and testValidMetadataSignatureWithoutIntermediateCaPublishedTrustingIntermediateCA.

Note : When adding a privatekey for metadata signing in the keystore, the privateKey needs to include its whole entityCertificate chain, it is not enough to add those separately.

3.14.19 Improved Cleanup of BouncyCastleBootstrapTest

Test that ensures the functionality of loading BouncyCastle as a Security provider would cause test failures in some environments. This was fixed by improving the Cleanup.

3.14.20 Test classes replacing application context should restore original context.

A clause to restore the original Application Context after replacing it for testing purposes has been added to the following test classes:

- AbstractLightMessageLoggerTest
- AUCONNECTORSAMLTest
- AUSERVICESAMLTest
- ColleagueRequestServletTest
- ColleagueResponseServletTest
- ConnectorIncomingEidasResponseLoggerFilterTest
- ConnectorIncomingEidasResponseLoggerTest
- ConnectorIncomingLightRequestLoggerFilterTest
- ConnectorIncomingLightRequestLoggerTest
- ConnectorMetadataGeneratorServletTest
- ConnectorOutgoingEidasRequestLoggerFilterTest
- ConnectorOutgoingEidasRequestLoggerTest
- ConnectorOutgoingLightResponseLoggerFilterTest
- ConnectorOutgoingLightResponseLoggerTest
- ProxyServiceIncomingEidasRequestLoggerFilterTest
- ProxyServiceIncomingEidasRequestLoggerTest
- ProxyServiceIncomingLightResponseLoggerFilterTest
- ProxyServiceIncomingLightResponseLoggerTest
- ProxyServiceMetadataGeneratorServletTest

- ProxyServiceOutgoingEidasResponseLoggerFilterTest
- ProxyServiceOutgoingEidasResponseLoggerTest
- ProxyServiceOutgoingLightRequestLoggerFilterTest
- ProxyServiceOutgoingLightRequestLoggerTest
- SecurityRequestFilterTest
- SecurityResponseHeaderHelperTest
- SpecificConnectorRequestServletTest
- SpecificProxyServiceResponseTest

3.14.21 Fix tests in EidasNodeMetadataGeneratorTest

The following fixes were done to fix tests in EidasNodeMetadataGeneratorTest:

- renamed method removeDir to tearDown
- removed method petMetadataInFile
- rewrote testGenerateMeadataConnectorasIdP and renamed it to testGenerateMetadata
- rewrote testGenerateMetadataWithContacts
- rewrote testGenerateMetadataWithOutSSOSPostLocation and renamed to testGenerateMetadataDecryptionCertificateException
- removed testGenerateMetadataWithOutSSOSRedirectLocation
- removed testGenerateMetadataWithoutSSOSRedirectLocationInpPropertiesFile (not a typo)
- removed testGenerateMetadataWithoutSSOSPosttLocationINpPropertiesFile (not a typo)
- moved long strings CONTACT_SOURCE and CONTACT_SOURCE_INCOMPLETE to private methods getContactSource and getContactSourceIncomplete for readability

3.14.22 Remove service configuration from EIDAS-Config eidas.xml other than CA and CB

As the demo implementation does not need more than two service configurations, the service configurations for CC through CF were removed in the following files:

- eidas.xml
- sp.properties
- dev.properties

3.14.23 Regression: HTTP error 500 on EidasNode/SpecificConnectorRequest when replaying the light request

The error message in case of a replayed light request has changed to 000011 - Invalid service provider Request. This is a fix as the previous (empty) message was not correct.

3.14.24 Improve Junit tests with NIST Curve P-256 cases

Since the introduction of the KeyAgreement in the Node's code, only JUnit tests with Brainpool curve were existing. The SAMLAuthnResponseKeyAgreementDecrypterTest class was modified to integrate unit tests for Nist curve.

Together with the adaptation of the class, two new tests files were added :

- saml_ec_NistCurve_kw-aes128_encrypted_response.xml
- saml_ec_NistCurve_kw-aes256_encrypted_response.xml

And the two previously existing tests file for Brainpool were renamed from :

- saml_ec_kw-aes128_encrypted_response.xml
    - to saml_ec_Brainpool_kw-aes128_encrypted_response.xml
- saml_ec_kw-aes256_encrypted_response.xml
    - to saml_ec_Brainpool_kw-aes256_encrypted_response.xml

3.14.25 Signature algorithm is not validated with its own whitelist

Previously the CEF eID Node was only checking that the signature algorithm to be used was part of the authorised signature algorithm from the specifications. in order to be more coherent, this was modified so the signature algorithm to be used can only be one of the configured signature algorithm in the whitelist. By default the signature algorithm whitelist will assume that all signature algorithm from the specifications are accepted. To achieve this change the *AbstractProtocolSigner* class was modified with the removal of the *public static String validateSigningAlgorithm(@Nullable String signatureAlgorithmName)* method (since this method is not used anymore) and the addition of the *private void validateSigningConfiguration(final SignatureConfiguration signatureConfiguration)* which will validate that the signature algorithm whitelist is not empty and that the signature algorithm is part of the signature algorithm whitelist. The *AbstractProtocolSignerTest* class was also modified to remove the tests that were testing the removed method. Two new tests *testAbstractProtocolSignerWithInvalidMetadataAlgorithm* and *testAbstractProtocolSignerWithInvalidSignatureAlgorithm* were added to validate the *validateSigningConfiguration* method implementation. Two new test testGenerateMetadataWithAlgorithmNotInWhitelist and testGenerateMetadataWithWhitelistEmpty where added to the Eidas-Node module to verify the algorithm whitelist functionality.

3.14.26 Use SHA-256 as Digest method with RSA-OAEP encryption

The Node was previously still using SHA1 as digest method when encrypting with RSA-OAEP, this has now been updated so the Node uses SHA-256 instead. However, in the case of a Connector, that should receive the encrypted response, only supporting eIDAS protocol version 1.1, then the digest method used will still be SHA-1. This is a measure put in place to allow backward compatibility with 1.4.x versions. Two new classes (LegacySha1DigestX509*Certificate* and LegacySha1Digest*X509Credential*) were created, to permit the duality of behavior without modifying the interfaces. Tests for these classes were added. Therefore, the *SAMLAuthnResponseEncrypter* class has been modified, the method build the KeyEncryptionParameters when using Key Transport has been modified to add some RSAOAEPParameters with the SHA-256 digest method. The *SAMLAuthnResponseKeyTransportEncrypterTest* was also modified to correctly validate the digest method used when encrypting with RSA-OAEP. The *AbstractProtocolEncrypter* and the *AbstractProtocolEngine* have been modified to handle the adaptive behavior of the digest method algorithm to use.

Some tests have also been added to verify this in the following classes :

- *SAMLAuthnResponseEncrypterTestConfig*
- *SAMLAuthnResponseKeyTransportEncrypterTest*
- *AbstractProtocolEncrypterTest*
- *AbstractProtocolEngineTest*

And a new ProtocolProcessor configuration, called *"ProtocolVersioningWithCipher"* was added in the test *SamlEngine.xml* file as well as a new test resource file "MetadataFetcher_ProtocolVersioningWithCipher.xml".

3.14.27 Removed code smell from EIDAS-Specific Proxy Service

Code from the "translateSpecificResponse" was refactored to remove a possible NPE.Upgrade org.apache.santuario:xmlsec dependency to version 2.2.3

3.14.28 Fix javadocs errors due to Java 11

Since the upgrade to java 11, new errors appeared when generating the Javadoc.

Therefore, the following classes were modified to fix those javadoc generation errors :

- EIDAS-Commons/src/main/java/eu/eidas/auth/commons/IPersonalAttributeList.java
- EIDAS-Encryption/src/main/java/eu/eidas/encryption/exception/CertificateException.java
- EIDAS-Light-Commons/src/main/java/eu/eidas/auth/commons/attribute/impl/reflect/GenericTypeReflector.java
- EIDAS-Metadata/src/main/java/eu/eidas/engine/exceptions/EIDASMetadataException.java
- EIDAS-Metadata/src/main/java/eu/eidas/engine/exceptions/EIDASMetadataProviderException.java
- EIDAS-Metadata/src/main/java/eu/eidas/engine/exceptions/EIDASMetadataRuntimeException.java
- EIDAS-SAMLEngine/src/main/java/eu/eidas/auth/engine/core/validator/eidas/EidasValidator.java
- EIDAS-SAMLEngine/src/main/java/eu/eidas/engine/exceptions/EIDASSAMLEngineException.java
- EIDAS-SpecificCommunicationDefinition/src/main/java/eu/eidas/specificcommunication/exception/SpecificCommunicationException.java

3.14.29 Error page adaptation to include contact details

Error pages have been modified to include the country contact details ("connector.contact.support.email" and "service.contact.support.email") so users know who to contact in case of errors.

The following files were updated:

- EIDAS-Commons/src/main/java/eu/eidas/auth/commons/EidasParameterKeys.java
- EIDAS-Node/src/main/java/eu/eidas/node/logging/connector/ConnectorIncomingEidasResponseLoggerFilter.java
- EIDAS-Node/src/main/java/eu/eidas/node/logging/connector/ConnectorIncomingLightRequestLoggerFilter.java
- EIDAS-Node/src/main/java/eu/eidas/node/logging/service/ProxyServiceIncomingEidasRequestLoggerFilter.java
- EIDAS-Node/src/main/java/eu/eidas/node/logging/service/ProxyServiceIncomingLightResponseLoggerFilter.java
- EIDAS-Node/src/main/java/eu/eidas/node/NodeParameterNames.java
- EIDAS-Node/src/main/resources/errors.properties
- EIDAS-Node/src/main/webapp/interceptorError.jsp
- EIDAS-Node/src/main/webapp/internalError.jsp
- EIDAS-Node/src/main/webapp/presentError.jsp

3.14.30 (EID-1090) Decryption uses all keys in keystore to decrypt

EIDAS-Encryption will use the KeyInfo public key in order to select the correct key before decryping contents of a SAMLResponse.

Testcases added for PublicKey from X509Certificate and DEREncodedKeyValue.

The method decryptSAMLResponse(Response samlResponseEncrypted, X509Credential credential) was removed in SAMLAuthnResponseDecrypter.

Changes made in :

- SAMLAuthnResponseDecrypter
- SAMLAuthnResponseDecrypterTest
- SAMLAuthnResponseKeyAgreementDecrypterTest
- SAMLAuthnResponseKeyTransportDecrypterTest

3.14.31 Upgrade org.apache.santuario:xmlsec dependency to version 2.2.3

Due to a CVE impacting dependency org.apache.santuario:xmlsec and upgrade was done to use 2.2.3 not impacted version.

3.14.32 Invalid error page when redirect.binding is disallowed

Added an exception which results in an error page that will be shown when a request with "GET" method binding is received and "allow.redirect.binding" is set to false.

Changes were made in:

- EIDAS-Node/src/main/java/eu/eidas/node/service/ColleagueRequestServlet.java
- EIDAS-Node/src/main/java/eu/eidas/node/connector/SpecificConnectorRequestServlet.java

3.14.33 SAML response message not logged by ProxyServiceOutgoingEidasResponseLogger when Mandatory Attribute not found

Due to an issue with jsp filtering from Tomcat server, we decided to replace urlPatterns value from ProxyServiceOutgoingEidasResponseLoggerFilter with ServiceExceptionHandler servlet, for the server to filter it correctly.

3.14.34 Replace Cobertura plugin by JaCoCo plugin for Code coverage

Since the upgrade to java 11, the Cobertura plugin, which latest version is from 2015, was not working anymore. Therefore, it was replaced by the latest version (v. 0.8.7) of the JaCoCo maven plugin.