e-Trusted Exchange Platform

# Time Stamping and Signature support

# Scope of the document

The scope of the present document is to describe the support for time stamping and signature in eTrustEx platform.

# Presentation

E-TrustEx needs to support (for non-repudiation) time stamping and electronic signature of the acknowledgement message returned by its asynchronous services.

The acknowledgment generation has been refactored to provide flexibility on the acknowledgment generation on one side and on the acknowledgment signature on the other side.

The system relies on specific metadata:

* ACK\_GENERATOR\_CLASS: it defines the class to be used in order to generate the acknowledgment;
* ACK\_SIGNATURE\_CLASS: it defines the class to be used in order to sign the acknowledgment.

The class specified in ACK\_GENERATOR\_CLASS metadata must implement the interface eu.europa.ec.cipa.etrustex.integration.ack.IAckGenerator that defines a method generateAck.

A generic implementation currently used by all the asynchronous services is provided in the class eu.europa.ec.cipa.etrustex.integration.ack.DefaultAckGenerator.

The generation and content of the acknowledgment are outside the scope of this document.

More information on that topic can be found in UC1\_002 Process Asynchronous Message use case [see reference 1].

The signature and time stamping of the acknowledgement are implemented in a class specified by the metadata ACK\_SIGNATURE\_CLASS that must implement the interface eu.europa.ec.cipa.etrustex.integration.ack.IAckSignatureGenerator and in particular its method signMessage.

Currently a dummy implementation is provided in the class eu.europa.ec.cipa.etrustex.integration.ack.DefaultAckSignatureGenerator but when a concrete needs will come to sign and eventually time stamp acknowledgment with a given service provider like ESSI [see reference 2] at the European Commission or DSS [see reference 3] for an open source project (or any other implementation depending on the concrete needs) a specific implementation will be necessary.

Nevertheless eTrustEx has limited the impact of such support:

* its current code does not have to be modified;
* a specific class must be implemented and it just has to implement a simple interface;
* the new provided class must be configured in the metadata (dynamic configuration in the database).

# References

1. UC1\_002 Process Asynchronous Message: [https://webgate.ec.europa.eu/CITnet/svn/ETRUSTEX/trunk/001 e-TrustEx/002 Requirements/Use Case Model/New Use Cases/UC1\_002 Process Asynchronous Message.doc](https://webgate.ec.europa.eu/CITnet/svn/ETRUSTEX/trunk/001%20e-TrustEx/002%20Requirements/Use%20Case%20Model/New%20Use%20Cases/UC1_002%20Process%20Asynchronous%20Message.doc)
2. ESSI project: <https://webgate.ec.europa.eu/esig>
3. DSS project: <https://joinup.ec.europa.eu/asset/sd-dss/home>