D8.3

Conformance and Interoperability Testing Result Report (1)

Document Identification	
Date	29.08.2018
Status	Final
Version	1.0

Related WP	WP3, WP4, WP5, WP6	**Related Deliverable(s)**	D2.3, D2.14, D3.3, D3.4, D4.3, D4.4, D5.2
Lead Authors	TUBITAK	**Dissemination Level**	PU
Lead Participants	TUBITAK	**Contributors**	TUBITAK
Reviewers	CET, TIL		

This document is issued within the frame and for the purpose of the LIGHT^{est} project. LIGHT^{est} has received funding from the European Union’s Horizon 2020 research and innovation programme under G.A. No 700321.

1. Executive Summary

This document is the deliverable D8.3 “Conformance and Interoperability Testing Result Report (1)” of the project “Lightweight Infrastructure for Global Heterogeneous Trust management in support of an open Ecosystem of Stakeholders and Trust schemes” (LIGHT*est*, project nr. 700321) with the objective to create a global cross domain trust infrastructure that renders it transparent and easy for verifiers to evaluate electronic transactions.

LIGHT*est* develops a lightweight trust infrastructure providing parties of electronic transactions with automatic validation of trust based on their individual trust policies. To ease integration and improve the availability on any system, LIGHT*est* makes use of the existing global Domain Name System (DNS) for publication, querying, and cross-jurisdiction translation of information relevant to make such trust-related decisions, including levels of assurance. Building on top of the existing global infrastructure of the domain name system and explicit efforts to reach international acceptance enable LIGHT*est* to offer truly “global trust lists”.

WP8 aims at rendering all LIGHT*est* software components seamlessly integrated, mature and robust. For this purpose, WP8 takes advantage of the methodology, tools and the experience gained from the previous FP7 Large Scale Pilot projects such as e-SENS (e-SENS, 2017) and STORK2.0 (STORK2.0, 2017). Minder Testbed is used and adopted for the further advancements of the automated conformance and interoperability testing approach with respect to the LIGHT*est* project testing requirements within the scope of WP8. In addition, a code review will be conducted as a part of WP8 activities to improve the quality of LIGHT*est* code even more by following the approach of FP7 STORK2.0 Large Scale Pilot project.

Task T8.3 is dedicated to the conformance and interoperability testing of the LIGHT*est* components via the use of Minder Testbed. These tests will ensure that the LIGHT*est* components which are developed within the course of the project fulfill the requirements or a subset of these in case of a particular conformance profile or level. Moreover, this task also aims to verify that the LIGHT*est* components can work together meaningfully.
This document presents an overview of the testing architecture that will be followed throughout the project. Next, the testing methodology is provided by presenting the Minder Test Assertion Model and a short guideline on how to write test assertions. In addition, test assertions for TSPA, TTA, and DP are given. Results of the testing will be provided in future iterations of the report.

2. Document Information

2.1 Contributors

Name	Partner
Berkay TOPÇU	TUBITAK
Burçin Bozkurt GÜNAY	TUBITAK
Edona FASLLIJA	TUBITAK
Elif Üstündağ SOYKAN	TUBITAK
Emine BİRCAN	TUBITAK

2.2 History

Version	Date	Author	Changes
0.0	23/07/2018	Edona FASLLIJA	First Draft, TOC
0.1	10/08/2018	TUBITAK	Initial Version
1.0	29/08/2018	Elif Üstündağ SOYKAN	Update due to review comments

3. Table of Contents

3.1 Table of Figures

3.2 Table of Acronyms

API	Application Program Interface
ATV	Automatic Trust Verifier
CC	Conformance Clause
DNS	Domain Name System
DNSSEC	Domain Name System SECurity extensions
DP	Delegation Publisher
eIDAS	Electronic Identification, Authentication and trust (Services)
eT	Electronic transaction
FR	Functional Requirement
HTTP(S)	Hypertext Transfer Protocol (Secure)
ISTQB	International Software Testing Qualifications Board
MTDL	Minder Test Definition Language
M1	Minder END User ATV Adapter
M2	Minder ATV Adapter
NS	Normative Statement
OASIS	Advancing Open standards for information society
PDF	Portable Document Format
RA	Reference Architecture
PTR	Pointer
REST	Representational State Transfer (service)
RR	Resource Record
S/MIME	Secure/Multipurpose Internet Mail Extensions
SUT	System Under Test
TA	Test Assertion
TP	Trust Policy
TPL	Trust Policy Language
TSLTS	Trust Service Status List Technical Specification
TSL	Trust Service Provider Trust Service Status List
TSP	Trust Service Provider
TSPA	Trust Scheme Publication Authority
TTA	Trust Translation Authority
URI	Uniform Resource Identifier
XML	Extensible Markup Language

4. Scope of the deliverable

4.1 Overview

The overall focus of the LIGHT*est* project is to develop a lightweight trust infrastructure providing parties of electronic transactions with automatic validation of trust based on their individual trust policies. By using an existing infrastructure of the global Domain Name System (DNS) for publication, querying, and cross-jurisdiction translation of information relevant to make such decisions, including levels of assurance, LIGHT*est* aims to enable the use of truly “global trust lists”. With this approach LIGHT*est* will basically provide an infrastructure to realize the most important principles and driving factors of eIDAS on a global level.

Conformance testing, also known as compliance testing, is a methodology used in software engineering to ensure that a product, process, computer program or system meets a defined set of standards. In this task, we will test outputs of other WPs in order to see whether they conform to the proposed specifications and standards. Interoperability testing, on the other hand, verifies whether all the systems exchange and use information properly, interpret the exchanged information meaningfully, and multiple entities work together in a harmonious way.

This deliverable is structured as follows. Section 1 presents the executive summary. Section 2 basically includes document information and Section 3 gives the table of contents. Section 4 presents an overview of WP8 and scope of this deliverable. Section 5 gives a testing architecture overview. The testing methodology is discussed in Section 6. Section 7 presents the conformance and interoperability test assertion for TSPA, TTA, and DP. Finally, Section 8 concludes the deliverable.

4.2 Scope

Within the course of the LIGHTest project, conformance and interoperability testing for the software components developed in WP3, 4, 5, and 6 will be carried out by using Minder Testbed and the results will be reported periodically. D8.3 Conformance and interoperability testing report (1) is the first report of this series and focuses on conformance and interoperability testing of the LIGHTest components. Although the testing duties within the LIGHTest project include integration testing and code quality review within the WP8 activities, Task 8.3 is dedicated to the automatic testing of the LIGHTest components in terms of their conformity to the related specifications and standards. In addition, interoperability of the LIGHTest components with each other is addressed.

The main contents of this deliverable are test methodology, specifications from requirements and design document, and test assertions. The Conformance Tests scope is defined by the process of verifying that an implementation of a component conforms to its specifications. Besides the conformance tests, the implementations of the LIGHTest components must be interoperable: one’s outcome must be understandable by another. This is the scope of Interoperability tests. All the concepts that are presented and discussed in this document are carried out within the light of these definitions.

In order to assert that implementations of LIGHTest components conform to their respective specifications, WP8 needs to thoroughly test the components developed by WP3, 4, 5, and 6 until the implementation covers all the generic requirements of the component. This document provides an overview of the Minder Test Assertion Model and a short guideline on how to write test assertions. In addition, the conformance clause, the normative statements and the test assertions are provided to ensure that a TSPA, TTA or DP implementation conforms to corresponding specifications. Results of the testing will be provided in future iterations of the report.

5. Testing Architecture Overview

The LIGHTest reference architecture (RA) has been provided in Figure 1. The testing plan includes the inspection of the general system behavior after a full deployment. All the communication between the deployed components will be intercepted, inspected and modified if necessary, then forwarded to its actual destination by Minder. The scripting support on Minder will help a full automation of the tests, and the interception of the messages flowing between the components will be accomplished by the so-called Minder Adapters.

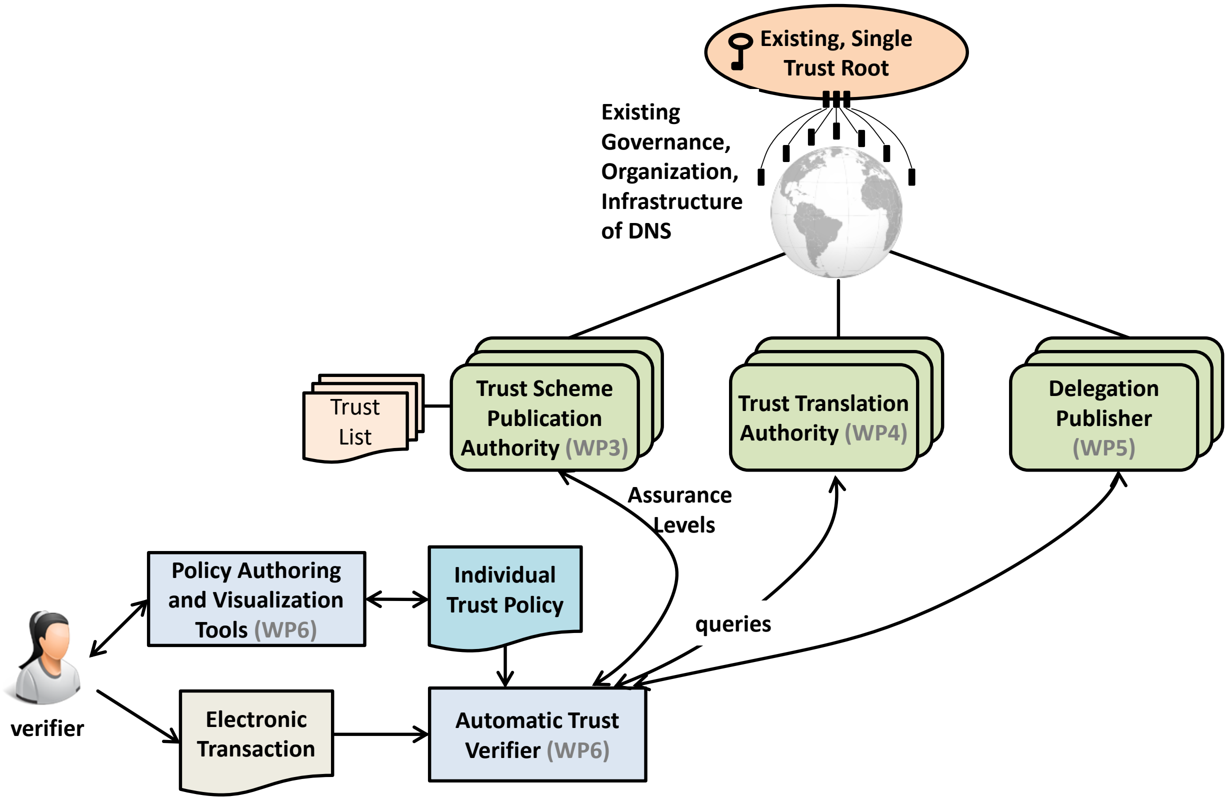
Existing, Single Trust Root
Existing Governance, Organization, Infrastructure of DNS
Trust List
Trust Scheme Publication Authority (WP3)
Trust Translation Authority (WP4)
Delegation Publisher (WP5)
Assurance Levels
queries
Policy Authoring and Visualization Tools (WP6)
Individual Trust Policy
verifier
Electronic Transaction
Automatic Trust Verifier (WP6)

Figure 1: The LIGHT*est* Reference Architecture

A Minder Adapter provides two aspects (views) in terms of interfaces. The first side interacts with the interfaces of system component being tested (i.e. SUT). This side ensures that the SUT does not know any detail about testing and 'believes' that it is sending its message to the actual target. In that sense, a Minder Adapter acts a 'target component'. The second aspect of the adapter interacts with Minder and converts the messages flowing within the system to the Minder signals so that Minder can understand and interpret them. For the Minder Testbed applied architecture overview please see Figure 2.

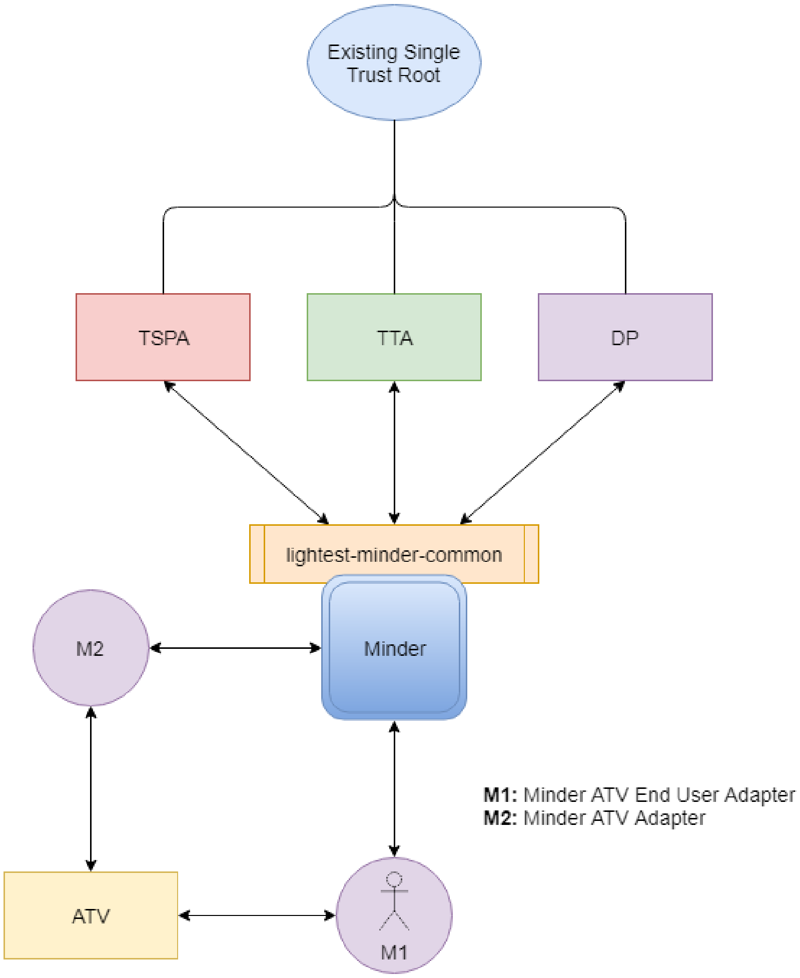

Figure 2 Minder Testbed Applied Architecture

The Minder adapters have been depicted as circular components in the architecture diagram. It wouldn't be hard to notice the interception of the links between the actual LIGHTest components. The messages flowing between Minder and the ATV component of LIGHTest is intercepted by the M1 and M2 adapters. Minder acts as a man-in-the-middle between the components. The communication between Minder and the TSPA, and TTA, can be handled directly via Minder's own DNS Client component, likewise, the communication between Minder and the DP can be handled via Minder's HTTP component eliminating the need for adapters for all these components.

A sample Minder adapter has been depicted in Figure 3. The ATVMinderAdapter, receives the DNS query that was issued by the ATV and converts it into a datagramRequestReceived signal call to Minder.

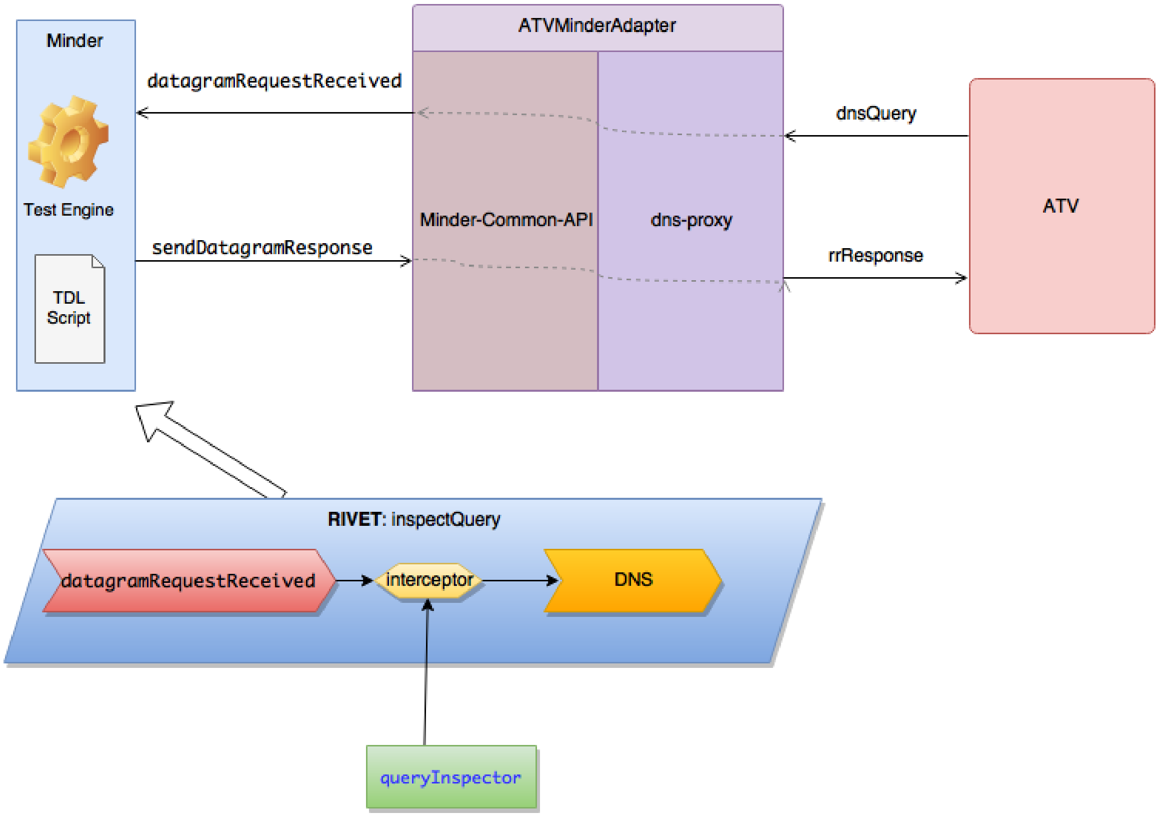

Figure 3 Sample Figure for Minder-TSPA Communication

At the arrival of the signal, TDL scripts for the corresponding Test Cases consisting of multiple rivets are executed. Each rivet can be seen as a single step of a test case that is run on the Minder Test Engine. The rivet enables the inspection of the query before being submitted to the LIGHTest DNS components. This is achieved through the queryInspector function, which takes a query having the DNS-like parameters, and returns it after the necessary inspections. The inspection might also involve the modification of the query if it is required in the test assertion corresponding to the test case. Minder uses Minder-common API and a DNS client API to convert the actual DNS queries and responses to signals/slots and vice versa.

The details of the Minder LIGHTest adapters such as the signals-slots and their parameters will be available in parallel to the development of the actual LIGHTest components and will be reflected in the Minder Testbed applied architecture.

6. Testing Methodology

As mentioned previously, one of the main objectives of WP8 is to assert that the implementations of LIGHTest components conform to their respective specifications.
For this purpose, WP8 needs to thoroughly test the components developed by the implementation Work Packages until the implementation covers all the generic requirements of the component.

The main steps of the testing methodology are as follows:

1. Generic Requirements – Generic Requirements that were developed at the start of the project (D2.3, WP2) are analyzed in order to identify potential specifications for the main LIGHTest components.
2. Generic Use Cases – Generic Use Cases described mainly in the LIGHTest reference architecture deliverable (D2.14) are studied in order to derive specifications related to the interoperability of the main components.
3. Conceptual Frameworks and Design Documents of Components – A further analysis of the design documents of each component's conceptual framework and design deliverables is carried out in order for the test cases to be up to date with the latest design details of the components.
4. Conformance and Interoperability Testing – In this step, we select the specifications that are in the scope of conformance and interoperability testing out of the pool of specifications that were gathered from the previous steps.
5. Test Assertions are written according to the OASIS Test Assertion Model, and the test cases for the test assertions are implemented and executed.

In this section we provide an overview of the Minder Test Assertion Model and a short guideline on how to write test assertions

6.1 Minder Test Assertion Model

The Minder Test Assertion Model is based on the OASIS Test Assertion Model (http://docs.oasis-open.org/tag/guidelines/v1.0/guidelines-v1.0.pdf).

The core parts of a test assertion according to this model, are listed below:

A **conformance clause (CC)** is a high-level statement that references normative sources:
Example Conformance Clause:
"An implementation conforms to the Component Requirements if it meets the following conditions: (Either other clauses or normative statements listed here)

A **normative statement (NS)** is a clear sentence that contains a subject, a may or must clause, and provides concise instructions (the word "SOME" is not used).

Example normative statement:

“A TSPA implementation MUST sign the URI Resource Records that are published in its DNS Name Server.”

A **target** is the implementation of the component that is referred. It is either the artifact or the tool to be tested. The target of a TA is the subject of the test. When the test assertion succeeds, we claim that the target has passed the test.

Example targets: TSPA, TTA, DP

A **prerequisite (optional)** is a logical statement that must be satisfied before the assertion. If the prerequisite evaluates to false, then the context is not valid for the current test assertion and the test is not done.

Example prerequisite for the normative statement given before:
“The received DNS Response is contains the _trust. prefix.”

A **predicate** is the logical statement that includes the assertion of one or more normative statements. It might be thought of as an **if** statement (in terms of a programming language). It evaluates to true or false.

Example predicate:
“The DNSSEC Validation of the Resource Record retrieved is successful.”

Figure 4 depicts the general anatomy of a Test Assertion, as described by OASIS and adopted by Minder.

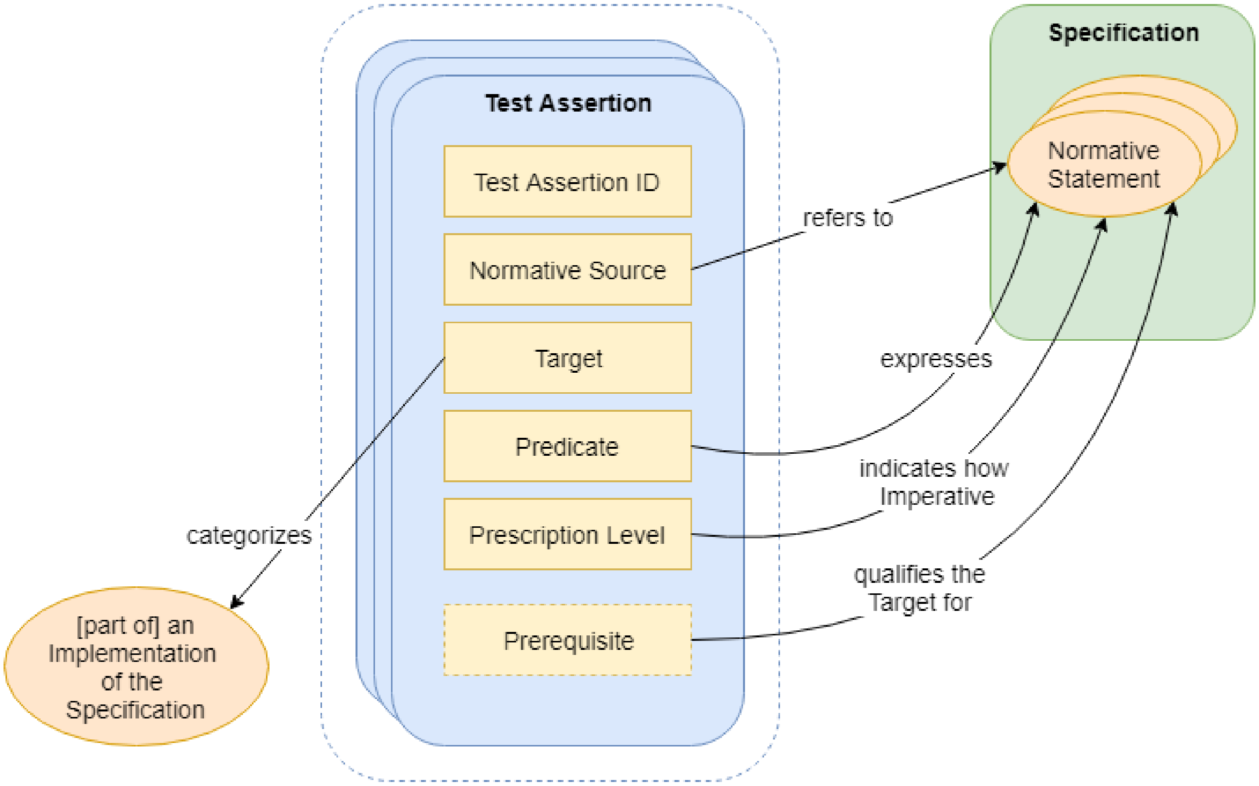

Figure 4 General Anatomy of a Test Assertion

6.2 Test Assertions Guideline

Test assertions are generated in the following steps:
(1) **Conformance clauses** are generated for a specification (e.g., a LIGHT^{est} component design specification).
(2) **Normative statements** are extracted for the conformance clauses.
(3) **Targets** are identified.
(4) **Prerequisites** are generated for the statements
(5) **Predicates** are generated for the normative statements.
(6) The predicates-prerequisites-normative statements are **formalized** using the OASIS format.

Figure 5 describes the main relationships between the concepts of Test Assertions, Normative Sources, and Test Cases.

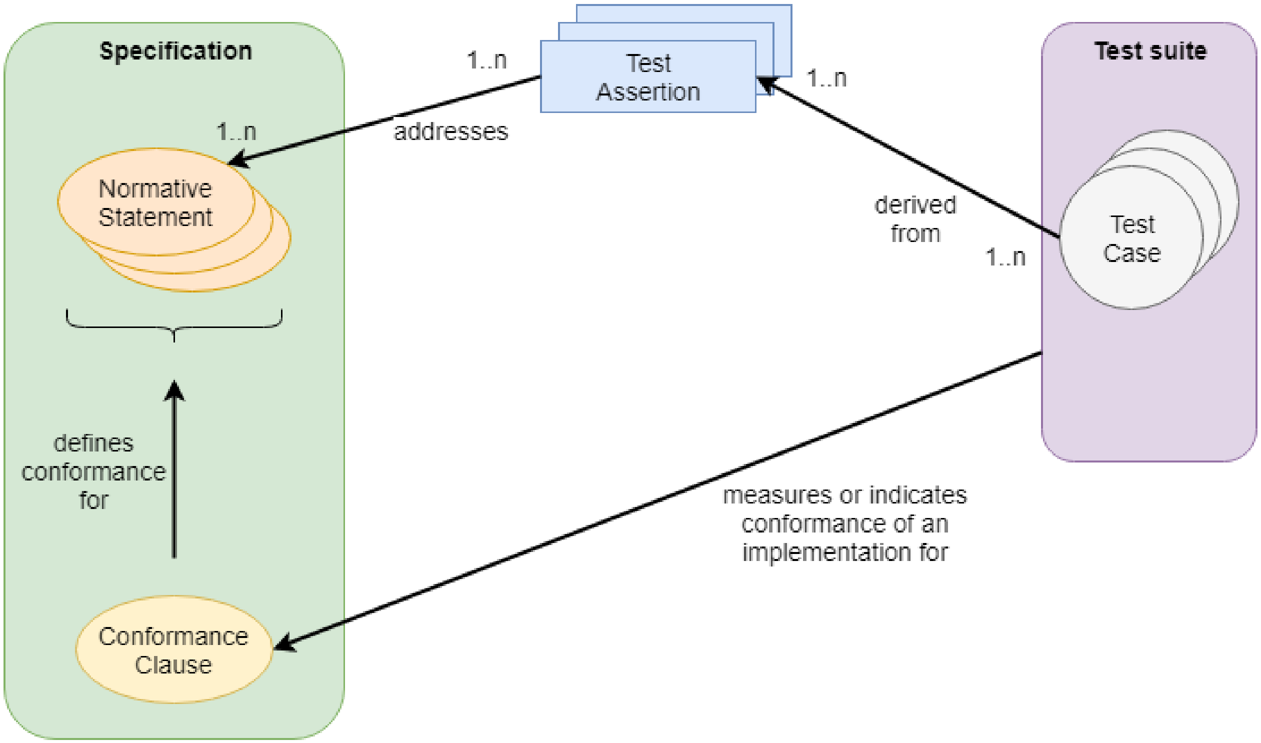

Figure 5 The relation between Test Assertions and Specifications[1].

7. Conformance and Interoperability Test Assertions

This section lists the first round of test assertions for TSPA, TTA, and DP, respectively. These test assertions were extracted from the respective Conceptual Framework and Design deliverables of these components. Following the methodology described in the previous section, we first list the normative sources together with their references, and then derive the test assertions to test for conformance to these specifications. The main categories that these specifications cover are Trust Declarations Publication, Discovery, Validity and Authenticity.

For traceability purposes, the identifiers of the items (conformance clause, normative sources and test assertions) was done using the following convention: **CC/NS/TA_LightestComponentName(TSPA,TTA,DP)_Number.**

7.1 TSPA

In order for a TSPA implementation to conform to TSPA Specifications, the conformance clause, the normative statements and the test assertions are provided below.

7.1.1 TSPA Conformance Clauses

We have only one conformance clause:

CC_TSPA_1: An implementation of TSPA is conforming to TSPA specifications (requirements, design) if it satisfies the conditions provided in the normative statements NS1-4.

The normative statements are provided as a sub-list of our conformance clause.

7.1.2 TSPA Normative Statements

NS ID	NS_TSPA_1
Reference	FR-05.00 - D2.3 Section 5.1
Description	The Trust Scheme Publication Authority MUST be able to operate an off-the-shelf DNS Name Server with the DNSSEC extension

NS ID	NS_TSPA_2
Reference	FR-05.01- D2.3 Section 5.1
Description	LIGHTest MUST be able to publish multiple Trust Lists under different sub-domains of the Authority domain name.

NS ID	NS_TSPA_3
Reference	FR-05.02 - D2.3 Section 5.1
Description	The utilities that parse selected Trust List formats MUST be able to be written or loaded into an equivalent DNS Zone files.

NS ID	NS_TSPA_4
Reference	D3.3 Section 5.3
Description	Subject Alternative Name field of the Certificate used to sign the electronic transaction points to the domain name (domain name used as Identifier) of the

	entity.

NS ID	NS_TSPA_5
Reference	D3.3 Section 5.3
Description	Issuer Alternative Name field of the Certificate used to sign the electronic transaction points to the domain name (domain name used as Identifier) of the issuer of the entity that issued the certificate.

NS ID	NS_TSPA_6
Reference	D3.3 Section 5.4
Description	The Authenticity of Trust Declarations of TSPA is ensured as follows: a. A URI Record is authentic if DNSSEC validation succeeds. b. A PTR Record is authentic if DNSSEC validation succeeds c. A Trust List is authentic a. if the certificate used to sign the list is valid under the constraints of the SMIMEA records published under the same domain name. b. if the signature is valid

NS ID	NS_TSPA_7
Reference	D3.3 Section 6.1
Description	A TSPA is composed of a DNS Name Server with DNSSEC extension that contains: a. Resource Records (PTR) for certificate Issuer (Issuer Name) – pointing to the URI RR of the Trust Scheme b. Resource Records (URI) for Trust Scheme (Scheme Name) – pointing to the Trust Scheme Provider c. Resource Records (SMIMEA) for Trust Scheme (Scheme Name)

NS ID	NS_TSPA_8
Reference	D3.3 Section 6.1
Description	A TSPA is composed of a HTTP Server (Trust Scheme Provider) that contains: a. Signed Trust Lists b. Tuple-based (ordinal and Boolean included) representations of Trust Schemes, provided as pointer from Trust List.

NS ID	NS_TSPA_9
Reference	D3.4 Section 5.3
Description	URI Resource Record contains exactly one URI as its record data.

NS ID	NS_TSPA_10
Reference	D3.4 Section 5.3
Description	PTR Resource Record contains another domain as its record data.

NS ID	NS_TSPA_11
Reference	D3.4 Section 9.1.2
Description	SMIMEA Resource Record contains four possible constraint fields, such as:

	a. CA constraints b. Service Certificate Constraints c. Trust Anchor Assertion d. Domain-Issued Certificate

NS ID	NS_TSPA_12
Reference	D3.4 – Section 6.1
Description	Trust Schemes published in the DNS Name Server of TSPA can either be Boolean or Ordinal.

NS ID	NS_TSPA_13
Reference	D3.4 Section 6.1
Description	IssuerName Query must be of the following structure: _scheme._trust.IssuerDomainName IN PTR

NS ID	NS_TSPA_14
Reference	D3.4 Section 6.2
Description	According to the format of the Trust Scheme, the Response to the IssuerName Query is a PTR Resource Record that contains : a. the domain name of the SchemeName if Boolean b. levelName.domainName of the SchemeName if Ordinal

NS ID	NS_TSPA_15
Reference	D3.4 Section 6.2
Description	SchemeName Query utilizes the Domain Name of the SchemeName obtained from NS_TSPA_12 and must be of the following structure: _scheme._trust.SchemeNameDomainName IN URI

NS ID	NS_TSPA_16
Reference	D3.4 Section 6.2
Description	The Response to the SchemeName Query is a URI Resource Record that points to the Trust Scheme Provider under which the trust list of the Scheme is published.

NS ID	NS_TSPA_17
Reference	D3.4 Section 6.2
Description	The response of a Trust List query (IssuerName_SchemeName_association query) MUST be a signed Association Statement/signed trust list (boolean)

NS ID	NS_TSPA_18
Reference	D3.4 Section 9.1.2
Description	A Certificate Constraint Query must be of the following structure: _scheme._trust. SchemeNameDomainName IN SMIMEA

7.1.3 TSPA Test Assertions

TA ID	TA_TSPA_1
Normative Source	NS_ TSPA_1
Target	TSPA
Prerequisite	
Prescription Level	Mandatory
Predicate	IP address of the TSPA DNS server exists and can be listed on the configurations and is already set on the TCP/IP Properties (DNS Server Address settings)

TA ID	TA_TSPA_2
Normative Source	NS_ TSPA_1
Target	TSPA
Prerequisite	The TSPA DNS Name Server is up and running and contains published trust scheme membership declarations.
Prescription Level	Mandatory
Predicate	The RR responses to the TSPA scheme membership query (IssuerName, SchemeName, CertificateConstraints) are signed by a valid Zone Key.

TA ID	TA_TSPA_3
Normative Source	NS_TSPA_6, NS_TSPA_7
Target	TSPA
Prerequisite	The TSPA DNS Name Server is up and running and contains published trust scheme membership declarations in the form of PTR Records. The ATV has issued an IssuerName query to the TSPA.
Prescription Level	Mandatory
Predicate	The RR response to the IssuerName query is a PTR Record and its DNSSEC validation is successful.

TA ID	TA_TSPA_4
Normative Source	NS_TSPA_6, NS_TSPA_7, NS_TSPA_16
Target	TSPA
Prerequisite	The TSPA DNS Name Server is up and running and contains published scheme locations declarations in the form of URI Records. The ATV has issued a SchemeNameLocation query to the TSPA.
Prescription	Mandatory

Level	
Predicate	The RR response to the SchemeNameLocation query is a URI Record and its DNSSEC validation is successful.

TA ID	TA_TSPA_5
Normative Source	NS_TSPA_6, NS_TSPA_7, NS_TSPA_17
Target	TSPA
Prerequisite	The TSPA DNS Name Server is up and running and contains published trust list declarations in the form of signed trust lists. The ATV has issued an IssuerName_SchemeNameAssociation query to the TSPA.
Prescription Level	Mandatory
Predicate	The RR response to the IssuerName_SchemeNameAssociation query is a signed association and its signature validation is successful.

TA ID	TA_TSPA_6
Normative Source	NS_TSPA_6, NS_TSPA_7
Target	TSPA
Prerequisite	The TSPA DNS Name Server is up and running and contains published trust list declarations in the form of signed trust lists. The ATV has issued an SchemeNameTuples query to the TSPA.
Prescription Level	Mandatory
Predicate	The RR response to the SchemeNameTuples query is a set of tuples retrieved from the pointer of the respective trust list entry.

TA ID	TA_TSPA_7
Normative Source	NS_TSPA_13
Target	TSPA
Prerequisite	The TSPA DNS Name Server is up and running and contains published trust scheme membership declarations in the form of PTR Records. The ATV has issued an IssuerName query to the TSPA.
Prescription Level	Mandatory
Predicate	The received DNS query is of the form _scheme._trust.IssuerDomainName IN PTR

TA ID	TA_TSPA_8
Normative Source	NS_TSPA_15, NS_TSPA_9
Target	TSPA

Prerequisite	The TSPA DNS Name Server is up and running and contains published trust scheme membership declarations in the form of PTR Records. The ATV has issued an SchemeNameLocation query to the TSPA.
Prescription Level	Mandatory
Predicate	The received DNS query is of the form _scheme._trust.SchemeNameDomainName IN URI

TA ID	TA_TSPA_9
Normative Source	NS_TSPA_18
Target	TSPA
Prerequisite	The TSPA DNS Name Server is up and running and contains published trust scheme membership declarations in the form of PTR Records. The ATV has issued a CertificateConstraints query to the TSPA.
Prescription Level	Mandatory
Predicate	The received DNS query is of the form _scheme._trust. SchemeNameDomainName IN SMIMEA

TA ID	TA_TSPA_10
Normative Source	NS_TSPA_14, NS_TSPA_12, NS_TSPA_10
Target	TSPA
Prerequisite	The TSPA DNS Name Server is up and running and contains published trust scheme membership declarations in the form of PTR Records. The ATV has issued an IssuerName query to the TSPA.
Prescription Level	Mandatory
Predicate	The RR response to the IssuerName query is a PTR Record containing the domain name of the SchemeName if the queried trust scheme is Boolean.

TA ID	TA_TSPA_11
Normative Source	NS_TSPA_14, NS_TSPA_12, NS_TSPA_10
Target	TSPA
Prerequisite	The TSPA DNS Name Server is up and running and contains published trust scheme membership declarations in the form of PTR Records. The ATV has issued an IssuerName query to the TSPA.
Prescription Level	Mandatory
Predicate	The RR response to the IssuerName query is a PTR Record containing levelName.domainName of the SchemeName if the

	queried trust scheme is Ordinal.

7.2 TTA

In order for a TTA implementation to conform to TTA Specifications, that were derived and built off from the guidelines and necessary functions defined for each component defined in the Reference Architecture (D2.14), the conformance clause, the normative statements and the test assertions are provided below.
A TTA is composed of two functionalities: a Trust Translation List Provider and a Trust Translation Publisher. [D4.3/5]

7.2.1 TTA Conformance Clauses

We have only one conformance clause:

CC_TTA_1: An implementation of TTA is conforming to TTA if it satisfies the conditions provided in the normative statements (NS_TTA_1-15) under the next section 7.2.2

7.2.2 TTA Normative Sources

Normative Sources are derived from D2.3 that includes the functional requirements and use cases defined for each component defined in D2.14

NS ID	NS_TTA_1
Reference	FR-06.00- TTA: Integratable with DNSSEC
Description	A Trust Translation Authority MUST operate a standard DNS Name Server with DNSSEC extension

NS ID	NS_TTA _2
Reference	FR-06.01- TTA: Trust Data Flexibility
Description	The TTA MAY publish multiple Trust Translation Lists under different subdomains.

NS ID	NS_TTA_3
Reference	FR-06.02- TTA: Utilities to Load selected Trust Translation Data
Description	Trust Translation Lists MUST be loaded into DNS Zone Files.

NS ID	NS_ TTA_4
Reference	FR-06.03- TTA: Formats, FR-07.07- TTA: Interface, D4.4 Section 7.1, D4.4 Section 7.2, D4.4 Section 7.3
Description	TTA (The Trust Translation Publisher&Authority) MUST be able to accept trust translation lists of all the required formats, such as Boolean, ordinal, and tuple-based.

NS ID	NS_ TTA_5
Reference	FR-06.06- TTA: Discoverability
Description	The translation lists MUST be discoverable through DNS according to the required URL formats.

Other Normative Sources extracted from the WP4 Documents are given below:

NS ID	NS_ TTA_6
Reference	D4.4 Section 7.1
Description	TTA module provides a trust translation file for each recognized trust level with either XML, TPL, or both

NS ID	NS_ TTA_7
Reference	D4.4 Section 7.3, NS_TTA_3, NS_TTA_1
Description	TTA will publish a pointer to the trust translation list for the group in the form of a series URI resource records

NS ID	NS_ TTA_8
Reference	D4.4 Section 6.2, NS_TTA_1, NS_TTA_3, NS_TTA_5
Description	Authenticity of Trust Translation List declarations is ensured as follows: a. A URI Record is authentic if DNSSEC validation succeeds. b. Trust Translation List is authentic a. if the certificate used to sign the list is valid under the constraints of the SMIMEA records if the signature is valid

NS ID	NS_TTA_9
Reference	D4.4 Section 6.2, NS_TTA_1, NS_TTA3, NS_TTA_5
Description	A TTA is composed of a DNS Name Server with DNSSEC extension that contains: a. Resource Records (URI) for Trust Scheme (Scheme Name) – pointing to the Translation Lists with either XML or TPL format Resource Records (SMIMEA) for Trust Translation Lists for the Trust Scheme (Scheme Name)

NS ID	NS_TTA_10
Reference	NS_TTA_6, D4.4 Section 6.1
Description	A TTA is composed of a public Rest API (HTTP Server (Trust Translation Provider)) that contains Signed Trust Translation Lists

NS ID	NS_TTA_11
Reference	D4.4 Section 7.1, NS_TTA_10
Description	The TTA provides a separate trust translation list file for each recognized trust level.

NS ID	NS_TTA_12
Reference	NS_TTA_11, NS_TTA_6
Description	In case of XML file type, TTA returns the list of the trust levels equivalents to the one requested with level name and trust scheme name

NS ID	NS_TTA_13
Reference	NS_TTA_11, NS_TTA_6
Description	In case of TPL file type, TTA should return the list of the trust levels equivalents to the one requested with level name, trust scheme name and TPL description

NS ID	NS_TTA_14
Reference	D4.4 Section 6.2
Description	Trust translation list documents, XML or TPL formats, are signed by the TTA with X.509 certificates.

NS ID	NS_TTA_15
Reference	D4.4 Section 6.2, NS_TTA_1
Description	TTA-DNS should provide certificate constraints to use for the verification of the translation list signature.

7.2.3 TTA Test Assertions

TTA Test Assertions are associated with CC_TTA_1 conformance clause in order to define the conformance specifications and later on test cases will be derived from test assertions that address the normative statements of the specification.

TA ID	TA_TTA_1
Normative Source	NS_TTA_1, NS_TTA_9
Target	TTA-DNS
Prerequisite	The name and details (characteristics) of the trust scheme are defined in the TSPA and received from TSPA
Prescription Level	Mandatory
Predicate	Depending on the operating system that TTA is working on, the IP address of the DNS server exists and can be listed on the configurations and is already set on the TCP/IP Properties (DNS Server Address settings).

TA ID	TA_TTA_2
Normative Source	NS_TTA_1, NS_TTA_9
Target	TTA-DNS
Prerequisite	TA_TTA_1 The name and details (characteristics) of the trust scheme are defined in the TSPA and received from TSPA
Prescription Level	Mandatory
Predicate	TTA-DNS provides a means to secure DNS data by using digital signatures and public key cryptography.

TA ID	TA_TTA_3
Normative Source	NS_TTA_1, NS_TTA_3, NS_TTA_5, NS_TTA_7
Target	TTA-DNS
Prerequisite	The TTA DNS Name Server is up and running and contains published trust translation list declarations.

Prescription Level Mandatory

Predicate For Boolean trust scheme, the prefixes for the TTA DNS record is set as "_translate" for the aspect and "_trust" for the application with the following format:

```
;; QUESTION SECTION: Client/ATV to the TTA
;_translate._trust.etimestamp.eidas.eu.  IN
URI

;; ANSWER SECTION: from the TTA
_translate._trust.etimestamp.eidas.eu.   IN
URI
               https://lightest.eu/ttl_qualif
               iedTimestampEidas1.tpl
_translate._trust.etimestamp.eidas.eu.   IN
URI
               https://lightest.eu/ttl_qualif
               iedTimestampEidas1.xml
               …
```

TA ID TA_TTA_4

Normative Source TA_NS_1, TA_NS_3, NS_TTA_4, NS_TTA_5, NS_TTA_7

Target TTA-DNS

Prerequisite The TTA DNS Name Server is up and running and contains published trust translation list declarations.
The names of the assurance levels just published by the TSPA have to be already retrieved from the TSPA by ATV

Prescription Level Mandatory

Predicate For Ordinal&Tuple Trust Scheme, the prefixes for the TTA DNS record are set as "_translate" for the aspect and "_trust" for the application and the assurance level (obtained from TSPA for the trust scheme) with the following format:

```
;; QUESTION SECTION: Client/ATV to the TTA
;_translate._trust.qualified.eseal.eidas.eu.  IN  URI

;; ANSWER SECTION: from the TTA
_translate._trust.qualified.eseal.eidas.eu.   IN  URI
                      https://lightest.eu/ttl_qualifiedSealEid
                      as1.tpl
                      …
_translate._trust.qualified.eseal.eidas.eu.   IN  URI
                      https://lightest.eu/ttl_qualifiedSealEid
                      asN.tpl

_translate._trust.qualified.eseal.eidas.eu.   IN  URI
                      https://lightest.eu/ttl_qualifiedSealEid
                      as1.xml
                      …
_translate._trust.qualified.eseal.eidas.eu.   IN  URI
```

	https://lightest.eu/ttl_qualifiedSealEidasN.xml

TA ID	TA_TTA_5
Normative Source	TA_NS_1, NS_TTA_2, TA_NS_3
Target	TTA-Trust Translation Publisher
Prerequisite	The TTA DNS Name Server is up and running and contains published trust translation list declarations.
Prescription Level	Mandatory
Predicate	TTA publishes a pointer to the trust translation list for the group in the form of a series URI resource records

TA ID	TA_TTA_6
Normative Source	NS_TTA_2, NS_TTA_6, NS_TTA_10, NS_TTA_11
Target	TTA-DNS
Prerequisite	The TTA DNS Name Server is up and running and contains published trust translation list declarations.
Prescription Level	Preferred
Predicate	TTA-DNS lookup result should include more than one file for each recognized trust level with XML or TPL format.

TA ID	TA_TTA_7
Normative Source	NS_TTA_2, NS_TTA_4, NS_TTA_10, NS_TTA_11
Target	TTA
Prerequisite	The TTA DNS Name Server is up and running and contains published trust translation list declarations. Boolean/Ordinal/Tuple based scheme declarations should be defined in TSPA.
Prescription Level	Preferred
Predicate	TTA provides trust translation lists for each recognized trust level with Boolean, Ordinal or Tuple trust scheme types.

TA ID	TA_TTA_8
Normative Source	NS_ TTA_2, NS_ TTA_6, NS_TTA_10, NS_TTA_11
Target	TTA-Trust Translation Provider

Prerequisite	The TTA DNS Name Server is up and running and contains published trust translation list declarations.
Prescription Level	Mandatory
Predicate	Trust Translation Provider provides a file for each recognized trust level with either XML, TPL, or both for the trust scheme.

TA ID	TA_TTA_9
Normative Source	NS_TTA_2, NS_TTA_10, NS_TTA_11
Target	TTA-Trust Translation Provider
Prerequisite	The TTA DNS Name Server is up and running and contains published trust translation list declarations. Trust translation lists are already defined in the XML or TPL format for the trust schemes
Prescription Level	Permitted
Predicate	TTA could provide more than one translation schemes for the trust scheme.

TA ID	TA_TTA_10
Normative Source	NS_TTA_2, NS_TTA_6, NS_TTA_10, NS_TTA_12
Target	TTA-Trust Translation Provider
Prerequisite	The TTA DNS Name Server is up and running and contains published trust translation list declarations. Trust translation lists are already defined in the XML format for the trust schemes The names of the assurance levels just published by the TSPA have to be already retrieved from the TSPA by ATV, in order to build the right domain name for asking for the translation.
Prescription Level	Preferred
Predicate	In case of XML, TTA returns the list of the trust levels equivalents to the one requested with level name and trust scheme name.

TA ID	TA_TTA_11
Normative Source	NS_TTA_2, NS_TTA_6, NS_TTA_10, NS_TTA_13
Target	TTA-Trust Translation Provider
Prerequisite	The TTA DNS Name Server is up and running and contains published trust translation list declarations. Trust translation lists are already defined in the TPL format for the trust schemes The names of the assurance levels just published by the TSPA have to be already retrieved from the TSPA by ATV, in order to build the right domain name for asking for the translation.

Prescription Level	Preferred
Predicate	In case of TPL, TTA returns the list of the trust levels equivalents to the one requested with level name, trust scheme name and TPL description.

TA ID	TA_TTA_12
Normative Source	NS_TTA_1, NS_TTA_3, NS_TTA_4, NS_TTA_8, NS_TTA_9
Target	TTA
Prerequisite	The TTA DNS Name Server is up and running and contains published trust translation list declarations.
Prescription Level	Mandatory
Predicate	For Boolean trust scheme, TTA-DNS checks whether the certificate used for signing the translation files is valid according with the content of DNS-SMIMEA resource record. ;; QUESTION SECTION: Verifying authenticity ;_translate._trust.etimestamp.eidas.eu. IN **SMIMEA** ;; ANSWER SECTION: _translate._trust.etimestamp.eidas.eu. IN SMIMEA <SMIMEA record data>

TA ID	TA_TTA_13
Normative Source	NS_TTA_1, NS_TTA_3, NS_TTA_4, NS_ TTA_8, NS_TTA_9
Target	TTA
Prerequisite	The TTA DNS Name Server is up and running and contains published trust translation list declarations.
Prescription Level	Mandatory

Predicate	For Ordinal&Tuple trust scheme, TTA-DNS checks whether the certificate used for signing the translation files is valid according with the content of DNS-SMIMEA resource record including the trust scheme and level of assurance
	`;; QUESTION SECTION: Verifying authenticity`
	`;_translate._trust.`**`qualified`**`.eseal.eidas.eu.   IN   `**`SMIMEA`**
	
	`;; ANSWER SECTION:`
	`_translate._trust.qualified.eseal.eidas.eu.   IN   SMIMEA   <SMIMEA record data>`

TA ID	TA_TTA_14
Normative Source	NS_ TTA_6, NS_ TTA_8, NS_TTA_9, NS_TTA_14
Target	TTA

Prerequisite	The TTA DNS Name Server is up and running and contains published trust translation list declarations. Trust translation lists are already defined for the trust schemes The names of the assurance levels just published by the TSPA have to be already retrieved from the TSPA by ATV, in order to build the right domain name for asking for the translation.
Prescription Level	Mandatory
Predicate	Trust translation list documents, XML or TPL formats, are signed by the TTA with X.509 certificates.

TA ID	TA_TTA_15
Normative Source	NS_TTA_15
Target	TTA
Prerequisite	The TTA DNS Name Server is up and running and contains published trust translation list declarations. The TTA should return the signed trust translation lists
Prescription Level	Mandatory
Predicate	TTA-DNS should provide certificate constraints to use for the verification of the translation list signature.

7.3 DP

In order to test if the Delegation Provider (DP) implementation conforms to the DP specifications, conformance clauses, the normative statements and the test assertions are given below.

7.3.1 DP Conformance Clauses

CC_DP_1: An implementation of DP is conforming to DP if it satisfies the conditions provided in the normative statements NS_DP_1 to NS_DP_13.

7.3.2 DP Normative Statements

NS ID	NS_DP_1
Reference	D5.2 Section 9.2
Description	The DP receives and stores the signed and encrypted delegation from the Mandator and stores the delegation in the internal database along with the encryption key (in an encrypted format) for the delegation and some other meta-information.

NS ID	NS_DP_2
Reference	D5.2 Section 9.3
Description	Delegations should be encrypted in XML format based on the ETSI 19 621

NS ID	NS_DP_3

Reference	D5.2 Section 9.3
Description	When publishing a delegation DP MUST produce a receipt as a response to the Mandatory with the content still tob e defined XXXX (content is not defined in the D5.2)

NS ID	NS_DP_4
Reference	D5.2 Section 9.3
Description	Encrypted XML file can contain one or more delegations. Each block in the delegation is signed individually by the Mandator

NS ID	NS_DP_5
Reference	D5.2 Section 9.3
Description	DP MUST validate delegations both in publication and validation processes.

NS ID	NS_DP_6
Reference	D5.2 Section 9.3
Description	Delegation consist of Delegation Info and Delegation Data. Delegation Information contains an Version Identifier, Sequence Number and List Issue Date and Time field. The Delegation Data is divided into General and Metadata. The General part contains the Proxy; certificate of the delegate, and the Mandators information. The Metadata field is divided into Mandatory; containing a Date of Issuance, Validity Time, General Restrictions and a Scope of Empowerment; and Specific; which contains the Domain Specific Fields.

NS ID	NS_DP_7
Reference	D5.2 Section 9.3
Description	DP should possess a revocation list to check if the delegation is revoked. If so, DP should return a notification saying that the delegation is revoked.

NS ID	NS_DP_8
Reference	D5.2 Section 9.4
Description	DP MUST respond only one revocation query at the time.

NS ID	NS_DP_9
Reference	D5.2 Section 9.4.1
Description	DP MUST have a revoke command interface so that the Mandator can send a revocation publication request to DP.

NS ID	NS_DP_10
Reference	D5.2 Section 9.4.2
Description	DP MUST sign the revocation response with the certificate that is issued by Mandator for the revocation purpose.

NS ID	NS_DP_11
Reference	D5.2 Section 9.4.2
Description	Revocation query that is sent to DP MUST include hash of the delegation

	(delegation_id) to be queried for revocation status.

NS ID	NS_DP_12
Reference	D5.2 Section 9.4.2
Description	Revocation response MUST include the delegation that is given to DP, the certificates that is used to sign and all certificates to build the trust chain. for the revocation purpose. DP should possess a revocation list to check if the delegation is revoked. If so, DP should return a notification saying that the delegation is revoked.

NS ID	NS_DP_13
Reference	D5.2 Section 9.4.2
Description	The ATV is required to check that the DP indeed has a valid entry containing (amongst others) a hash of the delegation. This check is to ensure that the delegation has not been revoked by the mandator.

7.3.3 DP Test Assertions

TA ID	TA_DP_1
Normative Source	NS_DP_1
Target	DP
Prerequisite	Delegation is prepared by Mandator as a signed and encrypted delegation in XML format.
Prescription Level	Mandatory
Predicate	The DP MUST return a receipt to the delegation publication request after it verifies the delegation. The issued receipt contains all information about the delegation.
TA ID	TA_DP_2
Normative Source	NS_DP_1
Target	DP
Prerequisite	Delegation is prepared by Mandator as a signed and encrypted delegation.
Prescription Level	Mandatory
Predicate	The DP response to delegation publication request MUST be verified by means of XML schema verification.

TA ID	TA_DP_3
Normative Source	NS_DP_2
Target	DP
Prerequisite	Delegation is prepared by Mandator as a signed and encrypted delegation.
Prescription Level	Mandatory
Predicate	The delegation that is prepared by Mandator MUST conform to

	ETSI 19 621.

TA ID	TA_DP_4
Normative Source	NS_DP_3
Target	DP
Prerequisite	Delegation is prepared by Mandator as a signed and encrypted delegation.
Prescription Level	Mandatory
Predicate	The DP response to delegation publication request MUST conform to with the content (it is not defined in D5.2 yet).

TA ID	TA_DP_5
Normative Source	NS_DP_4
Target	DP
Prerequisite	Delegation file is needed
Prescription Level	Optional
Predicate	Encrypted XML delegation file MAY contain several delegation blocks.

TA ID	TA_DP_6
Normative Source	NS_DP_5
Target	DP
Prerequisite	Delegation file is needed
Prescription Level	Mandatory
Predicate	DP MUST validate delegations both in publication and validation processes.

TA ID	TA_DP_7
Normative Source	NS_DP_6
Target	DP
Prerequisite	Delegation file and encryption key is needed.
Prescription Level	Mandatory
Predicate	Content of the Delegation MUST conform D5.2, 9.3 Table 2

TA ID	TA_DP_8
Normative Source	NS_DP_7
Target	DP

Prerequisite	Revoked Delegation file is needed.
Prescription Level	Mandatory
Predicate	DP MUST return a notification saying that the delegation is revoked.

TA ID	TA_DP_9
Normative Source	NS_DP_7
Target	DP
Prerequisite	Valid Delegation file is needed.
Prescription Level	Mandatory
Predicate	DP MUST return a notification saying that the delegation is valid.

TA ID	TA_DP_10
Normative Source	NS_DP_8
Target	DP
Prerequisite	Revoked Delegation file is needed.
Prescription Level	Mandatory
Predicate	DP MUST return error if verifier sends more than one revocation query at the time.

TA ID	TA_DP_11
Normative Source	NS_DP_9
Target	DP
Prerequisite	Revoked Delegation file is needed. Mandator should delegate DP to revoke the respective delegation. To revoke a delegation, the Mandator must identify the delegation which delegation to revoke first.
Prescription Level	Mandatory
Predicate	DP MUST have a revoke command interface so that the Mandator can send a revocation publication request to DP.

TA ID	TA_DP_12
Normative Source	NS_DP_10
Target	DP
Prerequisite	Revoked Delegation file is needed. Mandator should delegate DP to revoke the respective delegation. To revoke a delegation, the Mandator must identify the delegation which delegation to revoke first.
Prescription	Mandatory

Level	
Predicate	DP MUST respond a revocation response when verifier sends a revocation query.

TA ID	TA_DP_13
Normative Source	NS_DP_10
Target	DP
Prerequisite	Revoked Delegation file is needed. Mandator should delegate DP to revoke the respective delegation. To revoke a delegation, the Mandator must identify the delegation which delegation to revoke first.
Prescription Level	Mandatory
Predicate	DP MUST sign the revocation response with the certificate that is issued by Mandator for the revocation purpose.

TA ID	TA_DP_14
Normative Source	NS_DP_11, NS_DP_13
Target	DP
Prerequisite	Revoked Delegation file is needed. Mandator should delegate DP to revoke the respective delegation. To revoke a delegation, the Mandator must identify the delegation which delegation to revoke first.
Prescription Level	Mandatory
Predicate	Revocation query that is sent to DP MUST include hash of the delegation (delegation_id) to be queried for revocation status. If delegation not found DP MUST return an error.

TA ID	TA_DP_15
Normative Source	NS_DP_12
Target	DP
Prerequisite	Revoked Delegation file is needed. Mandator should delegate DP to revoke the respective delegation. To revoke a delegation, the Mandator must identify the delegation which delegation to revoke first.
Prescription Level	Mandatory
Predicate	DP's response for revocation query MUST include the delegation that is given to DP, the certificates that is used to sign and all certificates to build the trust chain.

8. References

The LIGHTest Project, D2.3 – Requirements and Use Cases, Project Deliverable, 2017; https://www.lightest.eu//static/deliverables/D2.3.pdf

The LIGHTest Project, D2.14 - Reference Architecture, Project Deliverable, 2017; https://www.lightest.eu//static/deliverables/D2.14.pdf

The LIGHTest Project, D3.3 – DNS-based Publication of Trust Schemes, Project Deliverable, 2017; https://www.lightest.eu//static/deliverables/D3.3.pdf

The LIGHTest Project, D3.4 – Discovery of Trust Scheme Publication Authorities, Project Deliverable, 2017; https://www.lightest.eu//static/deliverables/D3.4.pdf

The LIGHTest Project, D4.3 – DNS-based Publication of Trust Translation Schemes, Project Deliverable, 2017; https://www.lightest.eu//static/deliverables/D4.3.pdf

The LIGHTest Project, D4.4 – Discovery of Trust Translation Authorities, Project Deliverable, 2017; https://www.lightest.eu//static/deliverables/D4.4.pdf

The LIGHTest Project, D5.2 – Conceptual Framework for Delegations (2), Project Deliverable, 2018.

9. Project Description

LIGHTest project to build a global trust infrastructure that enables electronic transactions in a wide variety of applications

An ever increasing number of transactions are conducted virtually over the Internet. How can you be sure that the person making the transaction is who they say they are? The EU-funded project LIGHTest addresses this issue by creating a global trust infrastructure. It will provide a solution that allows one to distinguish legitimate identities from frauds. This is key in being able to bring an efficiency of electronic transactions to a wide application field ranging from simple verification of electronic signatures, over eProcurement, eJustice, eHealth, and law enforcement, up to the verification of trust in sensors and devices in the Internet of Things.

Traditionally, we often knew our business partners personally, which meant that impersonation and fraud were uncommon. Whether regarding the single European market place or on a Global scale, there is an increasing amount of electronic transactions that are becoming a part of peoples everyday lives, where decisions on establishing who is on the other end of the transaction is important. Clearly, it is necessary to have assistance from authorities to certify trustworthy electronic identities. This has already been done. For example, the EC and Member States have legally binding electronic signatures. But how can we query such authorities in a secure manner? With the current lack of a worldwide standard for publishing and querying trust information, this would be a prohibitively complex leading to verifiers having to deal with a high number of formats and protocols.

The EU-funded LIGHTest project attempts to solve this problem by building a global trust infrastructure where arbitrary authorities can publish their trust information. Setting up a global infrastructure is an ambitious objective; however, given the already existing infrastructure, organization, governance and security standards of the Internet Domain Name System, it is with confidence that this is possible. The EC and Member States can use this to publish lists of qualified trust services, as business registrars and authorities can in health, law enforcement and justice. In the private sector, this can be used to establish trust in inter-banking, international trade, shipping, business reputation and credit rating. Companies, administrations, and citizens can then use LIGHTest open source software to easily query this trust information to verify trust in simple signed documents or multi-faceted complex transactions.

The three-year LIGHTest project starts on September 1st and has an estimated cost of almost 9 Million Euros. It is partially funded by the European Union's Horizon 2020 research and innovation programme under G.A. No. 700321. The LIGHTest consortium consists of 14 partners from 9 European countries and is coordinated by Fraunhofer-Gesellschaft. To reach out beyond Europe, LIGHTest attempts to build up a global community based on international standards and open source software.

The partners are ATOS (ES), Time Lex (BE), Technische Universität Graz (AT), EEMA (BE), G&D (DE), Danmarks tekniske Universitet (DK), TUBITAK (TR), Universität Stuttgart (DE), Open Identity Exchange (GB), NLNet Labs (NL), CORREOS (ES), IBM Danmark (DK) and Ubisecure (FI). The Fraunhofer IAO provides the vision and architecture for the project and is responsible for both, its management and the technical coordination.

The Fraunhofer IAO provides the vision and architecture for the project and is responsible for both, its management and the technical coordination.